

STATIC AND DYNAMIC ANALYSIS OF LINEAR ELASTIC SYSTEMS
OF NON-PRISMATIC THREE DIMENSIONAL BEAM ELEMENTS

by

L.N. da C. RESENDE

A Thesis submitted in partial fulfillment of the requirements for
the degree of Master of Science in the Faculty of Engineering
University of Cape Town

Department of Civil Engineering
UNIVERSITY OF CAPE TOWN

September, 1980.

The University of Cape Town has been given
the right to reproduce this thesis in whole
or in part. Copyright is held by the author.

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

(i)

DECLARATION OF CANDIDATE

I, Luis Nuno da Costa Resende, hereby declare that this thesis is my own work and that it has not been submitted for a degree at any other University.

Signed by candidate

September, 1980.

DEDICATION

I would like to dedicate this thesis to my parents and friends who encouraged me throughout this work.

ACKNOWLEDGEMENTS

I would like to extend my appreciation to Associate Professor W.S. Doyle^{*} under whose supervision this thesis was conducted, for his help, enthusiasm and the vital suggestions which made this work a reality.

My colleagues G.C. Howell and A.R. Lloyd with whom I had many very useful discussions.

The University of Cape Town and Joseph Stone Trust for their financial assistance.

Mrs. B. Atkinson for her immaculate typing of this document.

Mr. H. Cable for the printing of this thesis.

* Associate Professor, Department of Civil Engineering
University of Cape Town

ABSTRACT

A computer program, NONPRI [20], has been developed for the analysis of three dimensional skeletal assemblages consisting of non-prismatic members.

It is capable of static and dynamic analysis of structures consisting of members whose constitutive relationship is linear elastic.

The finite element formulation is based on the family of quadratic isoparametric finite elements. The three noded space frame element is quite versatile in that it can account for shear as well as flexural, axial and torsional deformation effects making it suitable for thin and thick beam analysis and for cases where the axial and torsional deformations are relevant. The element can be degenerated to a truss/frame transition element (3 translational degrees of freedom at each node - rotations ignored) and further degenerated to become a truss element. Furthermore, the element internal node is defined to lie at an arbitrary position inside the element. Thus, this flexibility in the non-prismatic element formulation makes it very powerful in practical analysis problems.

An out-of-core solution technique is used for the equations of static analysis bearing in mind the capability for solving large structural systems. An in-core solution technique is used for the equations of dynamic analysis bearing now in mind that these equations represent an iterative process which can otherwise become computationally very expensive.

The results obtained lead to the belief that the present formulation is very economical and accurate and consequently can become a useful analysis tool for a great variety of structural applications.

NOTATION

Upper Case Characters

A	Cross-sectional area
B, B_i	Strain matrix, submatrix referring to node i
C, C_i	Total, ith couple
C	Damping matrix
D	Stress matrix
E	Young's modulus of elasticity
F, F^e , F^s	Total, element, system equivalent nodal forces
G	Shear modulus
H, H_i	Matrix, submatrix of cross-sectional properties
I^e , I^s	Element, system matrix of coefficients
I_y , I_z	2nd moment of area about y, z axis
J	Torsional constant or Jacobian operator
K, K^e , K^s	Total, element, system stiffness matrix
K_i	ith stiffness matrix (in iterative procedure)
K_E , K_G	Elastic, geometrical stiffness matrix
L	Element length
M, M^s	Total, system mass matrix
M_i	ith mass matrix (in iterative procedure)
M_y , M_z	Bending moment about y, z axis
N, N_i	Shape function matrix, submatrix
N	Axial force
P_i	Load or Lagrangian polynomial
Q_y , Q_z	Shear force in y, z direction
S	Shear rigidity
S, S_e	Total, element surface area

Upper Case Characters (Continued)

T	Torsional moment
T^e, T_{ij}^e	Element transformation matrix, submatrix
V, V_e	Total, element volume
W	Work done
X, Y, Z	System cartesian axes set

Lower Case Characters

a_{ij}	Element of a matrix
f_r	r th principal minor
m	Semi-bandwidth of stiffness matrix
n	Order of stiffness, mass matrices
p	vector of body forces
p	Characteristic polynomial
q	Vector of surface tractions
q_u, q_ℓ	Number of eigenvalues associated with upper and lower limits of interval
u, v, w	Translational degree of freedom in x, y, z direction
x, y, z	Local cartesian axes set

Greek Characters

α_y, α_z	Warping constant for shear in y, z direction
γ	Torsional degree of freedom
$\delta, \delta_i, \delta^e, \delta^s$	Vector of, subvector of, element, system displacements
ϵ	Strain vector
θ_y, θ_z	Bending about y, z degrees of freedom
$\lambda, \lambda_j, \lambda_r$	Eigenvalue
Δ	$\text{diag}(\lambda_i)$ matrix

Greek Characters (Continued)

μ	Compressive stress factor
μ_k, μ_l, μ_u	kth, lower limit, upper limit shift
ν	Poisson's ratio
ξ	Natural coordinate
Π, Π_e	Total, element potential energy
ρ	Mass density
σ	Stress vector
ϕ_y, ϕ_z	Shear strain associated with y, z direction
ω	Natural frequency.

CONTENTS

DECLARATION	i
DEDICATION	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
NOTATION	vi
CONTENTS	viii
 CHAPTER 1 :	
INTRODUCTION AND BACKGROUND	1
 CHAPTER 2 :	
FINITE ELEMENT FORMULATION IN STATIC ANALYSIS	
2.1 Basic Equations for linear Equilibrium Problems	6
2.2 Element Characteristics and Assumptions	8
2.3 Element Definition	11
2.4 Natural/Element Cartesian Coordinate Systems Relationship	13
2.5 Strain/Displacement Relationship	14
2.6 Linear Elastic Stress/Strain Relationship	15
2.7 Element Stiffness Matrix Evaluation	16
2.8 Space Transformations and System Stiffness Matrix	16
2.9 Consistent Nodal Load Formulation	18
2.9.1 Point Concentrated Loads or Couples	19
2.9.2 Distributed Loads or Couples	20
2.10 Force and Moment Resultants	21
2.10.1 Optimal Sampling Points	21
2.10.2 Sampling Resultants at Integration Points	23
2.10.3 Sampling Resultants at External Nodes	23
2.10.4 Sampling Resultants at Internal Node	24
2.11 Interpolation Facilities	25
2.12 The Non-Prismatic Feature	26
2.13 The Element with Unevenly Spaced Nodes and Use of Lagrangian Interpolation Polynomials	26

2.14	Degenerate Forms of the Element	29
2.14.1	The Truss/Frame Transition Element	30
2.14.2	The Pure Truss Element	31
2 15	Modelling of Structures Using the NONPRI concept	32

CHAPTER 3 : SOLUTION OF EQUATIONS OF STATIC ANALYSIS - A FRONTAL APPROACH

3.1	Introduction	36
3.2	Choice of Solution Technique	36
3.3	General Description of the Frontal Solution Technique	37
3.4	Prefrontal Routine and suggested Modification	38
3.5	Formulation of Housekeeping Arrays	39
3.6	Assembly and Elimination Process	41
3.7	Back Substitution Phase	42
3.8	Remarks on the Frontal Solution Resolution Facility	43
3.9	Frontal Solution Operation	43
3.10	Advantages of the Frontal Solution Method in Static Analysis	43
3.11	Recent Developments of the Frontal Concept	45

CHAPTER 4 : FINITE ELEMENT FORMULATION IN DYNAMIC ANALYSIS

4.1	Basic Equations for Free Dynamic Vibration Problems	47
4.2	Element Characteristics and Assumptions	49
4.3	Element Definition	50
4.4	Cross-sectional Properties Matrix	50
4.5	Mass Density Matrix	51
4.6	Shape Functions Matrix	51
4.7	Schemes for Evaluating Element Mass Matrix	52

4.7.1	Lumped Matrix	43
4.7.2	Consistent Matrix	54
4.7.3	Diagonalised Consistent Matrix	54
4.8	Space Transformations and System Mass Matrix	55
4.9	A Basic Dynamic Condensation of Unwanted Degrees of Freedom	56

CHAPTER 5 : SOLUTION OF EQUATIONS OF DYNAMIC ANALYSIS - A STURM SEQUENCE APPROACH

5.1	Introduction	57
5.2	Review of Solution Methods for Eigenvalue Problems	58
5.2.1	Vector Iteration Methods	59
5.2.2	Transformation Methods	60
5.2.3	Polynomial Iteration Techniques	62
5.2.4	Methods Based on the Sturm Sequence Property	63
5.2.5	A Frontal Based Method with Static Condensation	65
5.3	Choice of Solution Method	67
5.4	The Adopted Sturm Sequence/Bisection/Inverse Iteration Algorithm	68
5.4.1	Sturm Sequence Property	69
5.4.2	The Bisection Procedure	69
5.4.3	The Computational Procedure	70
5.4.4	The Inverse Iteration Procedure	74
5.5	Scope and Advantages of Adopted Algorithm	75

CHAPTER 6 : NUMERICAL EXAMPLES AND RESULTS

6.1	Introduction	76
6.2	Static Analyses	76
6.2.1	Simply Supported Deep Beam	76
6.2.2	Tapered Cantilever	77

6.2.3	Three-Span Non-prismatic Beam	79
6.2.4	Grandstand Plane Frame Designed by Nervi	82
6.2.5	Three Dimensional Frame Highway Bridge Structure	83
6.2.6	Execution Times for Multi-storey Three Dimensional Frame	84
6.3	Dynamic Analyses	85
6.3.1	Cantilever Beam	85
6.3.2	Semicircular Arch	88
6.3.3	Multi-storey, Multi-bay Plane Frame	91
6.3.4	Unsymmetrical Space Frame	92
6.3.5	Tapered Cantilever	95
CHAPTER 7 :	CONCLUSIONS AND FURTHER DEVELOPMENTS	97
REFERENCES		100
APPENDIX A :	Third Node Method for Space Transformations	A-1
APPENDIX B :	Numerical Integration	B-1
APPENDIX C :	NONPRI Program Implementation	C-1
APPENDIX D :	NONPRI Program User Manual	D-1
APPENDIX E :	Sample Output from NONPRI	E-1

CHAPTER 1

INTRODUCTION AND BACKGROUND

Quadratic isoparametric finite elements are widely accepted and recognised as a basis for the analysis of two and three dimensional continuum structures [1, 2, 3, 4, 5]. However, the same attention has not been paid to the quadratic isoparametric line element for the analysis of skeletal structures.

At this stage it is perhaps convenient to note that the element herein developed using an isoparametric formulation need not have been formulated as isoparametric since it is a straight line element. In fact, a subparametric formulation would suffice since only 2 nodes are necessary to define the geometry of each element whereas 3 nodes are essential for the definition of its displacement variation. However, there are practical reasons for adopting an isoparametric formulation. The first of these reasons is that it is only necessary to use one set of interpolation or shape functions in the formulation making the computation 'neater'. Secondly, it facilitates further work when thinking in terms of upgrading the element to become a curved element. In essence, the element could still be classified as a subparametric one, and it should be realized that subparametric elements are in fact more often used in practice [1], especially in civil engineering applications. This can be visualized by thinking of instances when one needs to use curved elements. This is normally the case of curved boundaries in continuum structures where one wants to use a curved element to model the curved boundary accurately, but usually away from such boundaries one tends to use the more conventional straight sided element.

One of the aims of this work is then to present the development of such isoparametric line element and show how the quality of analysis can be improved when comparing it to the conventional

two-noded beam element explicit formulation.

Further, most of the generally available line elements do not take into account a possible variation of cross-section within the element and the option is to rather model a non-prismatic member as a discrete number of stepped constant property elements. The few line elements that do cater for variations of cross-sectional properties make use of some kind of exponential function approximation [6] for the variation along the axis of the element which is not always accurate since it requires knowledge of a taper parameter which varies for every different problem and also varies with the number of elements used in the modelling of a member. These parameters, necessary for an accurate solution, are therefore not always known or easy to establish.

It is again the aim of this work to go a step ahead and formulate the present three-noded element implicitly assuming that the cross-section properties vary along the element axis according to the same shape functions which represent the geometry and displacement fields, eliminating thus the need to establish any taper parameters.

As mentioned earlier the development of the present element is aimed at improving the quality of analysis. When one thinks of the conventional two-noded beam element in the context of static analysis it is immediately assumed that the displacement values and stress resultants are obtained at the two nodal points. This means that any discontinuities of, say, bending moment or shear force within the element cannot be 'picked up' and therefore wherever a discontinuity in the resulting quantities is expected to occur a new element must be started if an accurate solution is to be obtained. This leads, in many instances, to the use of an excessive number of elements in relation to the complexity of the structure. The existence of point loads or moments is an example of this condition. Further, if a coarse spacing of two-noded elements has been opted for, one might wish to calculate stress resultant values between the nodes. There is no automatic procedure for doing this when using a two-noded element.

The present three-noded element is designed to overcome all these difficulties. This is achieved by the arbitrary definition of the geometrical position of the internal node within the element. It allows one to place the internal node wherever (within certain limits for acceptable accuracy) a point effect (load, moment or prescribed displacement) occurs and therefore 'pick up' any possible discontinuities of resultant quantities within the element. The fact that in each half-element three values of each resultant quantity are sampled provides the facility of interpolating values anywhere along the element.

The degeneration of the present element to a truss/frame transition element (with 3 translational degrees of freedom at each node) and further down to a pure truss element is allowed making the element very useful and powerful from the point of view of practical application.

The versatility of the implicit isoparametric formulation means that accurate and more complete solutions can be obtained from a much fewer number of elements.

The importance of this fact is best realized in the context of dynamic analysis, which is often relevant in the design of structures. In particular, knowledge of the natural frequencies and associated modes of vibration is essential in assessing the dynamic response to external excitation, such as wind or earthquake forces in the case of civil engineering structures.

The adoption of consistent formulation techniques in the displacement method of analysis of vibrational problems has yielded an improvement in solution accuracy for a given level of structure idealisation in comparison with crude lumping procedures. Also, using the basic concepts of the Rayleigh-Ritz approach, it has been possible to guarantee the upper bound nature of the calculated results for frequency. These statements are valid if the structure is represented by elements of uniform cross-section [6]. When structures are composed of tapered members, the usual approach is

again to use the stepped representation mentioned previously. It is shown in this work that the use of relatively few elements in the stepped representation of a tapered member may give a much lower level of solution accuracy than in the case of a uniform member of the same length. The magnitude of this disparity is obviously associated with the severity of the taper. Furthermore, a lower bound solution is no longer achieved [6]. Thus, it appears desirable to formulate non-prismatic elements for the analysis of tapered members. A very practical and strong motivation for this approach is in the fact that dynamic analysis, due to its iterative nature, usually requires formidable computational effort and expense which can be drastically reduced by establishing mass and stiffness matrices of acceptable accuracy with the fewest number of elements and therefore degrees of freedom. This is achieved by the use of non-prismatic formulations. It should be noted that the advantages of the non-prismatic formulation can be offset if the computational effort associated with it is greatly in excess of that associated with uniform element formulations. For this reason, an approximate formulation using the same shape functions as for the geometry and displacement fields is used.

Another practical limitation in the stepped representation of non-prismatic members is in the substantially reduced accuracy of the solutions when a coarse 'grid' is used. This is not significant in the analysis of a beam with one or two spans where a refined 'grid' can and is normally used. However, in frameworks consisting of non-prismatic members a lot of care has to be taken in the 'grid' refinement if few elements are to be used for each tapered member.

The fact that relatively few elements are necessary to obtain a satisfactory answer to a particular problem when using the present formulation clearly indicates that a different attitude towards the modelling of structural systems must be taken by the analyst. This topic will be expanded upon as the argument develops.

Finally, it should be said that numerous numerical examples comprising both static and dynamic analysis problems have been used to test the

accuracy and effectiveness of the program NONPRI. The most relevant examples are presented in this work and the results are compared with known analytical solutions or solutions for other generally accepted programs. On the whole the results obtained from NONPRI are found to be very satisfactory.

CHAPTER 2

FINITE ELEMENT FORMULATION IN STATIC ANALYSIS

2.1 Basic Equations for Linear Equilibrium Problems

The finite element displacement method of structural analysis is here considered in detail. However, it should be realized that all equilibrium problems (ie. those in which the system does not vary with time) are similar in nature, the difference being reflected in the corresponding finite element models. Examples of equilibrium problems include stress analysis, electrostatics, magnetostatics, steady-state thermal conduction and fluid flow in porous media.

In the case of structural applications the governing equilibrium equations can be obtained by minimising the total potential energy of the system. The potential energy, Π , can be expressed as

$$\Pi = \frac{1}{2} \int_V [\sigma]^T \epsilon \, dV - \int_V [\delta]^T p \, dV - \int_S [\delta]^T q \, dS \quad (2.1)$$

where σ and ϵ are the stress and strain vectors respectively, δ the displacements at any point, p the body forces per unit volume and q the applied surface tractions. The integrations are taken over the volume V of the structure and over the loaded surface area S .

The first term on the right hand side of equation (2.1) accounts for the internal strain energy whereas the second and third terms represent the work contributions of the body forces and surface loads respectively.

In the finite element displacement method, the primary unknowns are the displacements at the chosen nodal points, so that the variation within an element is described in terms of nodal values by means of interpolation or shape functions. Therefore

$$\delta = N \delta^e \quad (2.2)$$

where N is the vector of interpolation functions and δ^e is the vector of nodal displacements. The strains within the element are expressed in terms of the element nodal displacements as

$$\epsilon = B\delta^e \quad (2.3)$$

where B is generally called the strain matrix and is composed of shape functions and their derivatives. Lastly, the stresses are related to the strains by some sort of constitutive relationship, as follows

$$\sigma = D\epsilon \quad (2.4)$$

where D , in this case, is the elasticity matrix.

Provided the element shape functions have been chosen so that no singularities exist in the integrands of the functional, the total potential energy of the structural system is the sum of the energy contributions of the individual elements. Thus,

$$\Pi = \sum_e \pi_e \quad (2.5)$$

where π_e is the potential energy of element e which, using equation (2.1) can be written as

$$\pi_e = \frac{1}{2} \int_{V_e} [\delta^e]^T [B]^T D B \delta^e dV - \int_{V_e} [\delta^e]^T [N]^T p dV - \int_{S_e} [\delta^e]^T [N]^T q dS \quad (2.6)$$

where V_e and S_e refer to element volume and loaded surface area.

Minimisation of the element potential energy with respect to nodal displacements yields

$$\begin{aligned} \frac{\partial \pi_e}{\partial \delta^e} &= \int_{V_e} ([B]^T D B) \delta^e dV - \int_{V_e} [N]^T p dV - \int_{S_e} [N]^T q dS \\ &= K \delta^e - F^e \\ &= 0, \end{aligned} \quad (2.7)$$

where

$$K^e = \int_{V_e} [B]^T D B dV \quad (2.8)$$

is the element stiffness matrix, and

$$F^e = \int_{V_e} [N]^T p dV + \int_{S_e} [N]^T q dS \quad (2.9)$$

are the equivalent nodal forces for the element.

From (2.7), one can write the structural system basic equilibrium equation as

$$K\delta = F \quad (2.10)$$

where K, δ, F are the system stiffness, displacements and applied forces respectively.

2.2 Element Characteristics and Assumptions

A straight line element with six degrees of freedom at each node (in its most general form) is considered. The cross-section is allowed to vary along the axis of the element, but the degenerate form of uniform cross-section is also possible.

A parabolic isoparametric formulation is adopted.

Unlike most of the more common types of beam element, the isoparametric beam formulation here considered can take account of transverse shear deformation since energy due to shear as well as bending is incorporated in the formulation. This makes the element capable of thin as well as thick beam analysis. It can also be used to analyse accurately three dimensional frames since energy due to axial and torsional deformation effects is included.

The main assumption made [7] is concerned with the cross-sectional behaviour of the element. Usually, in beam theory it is assumed that

The potential energy of the element can then be expressed as

$$\begin{aligned}\Pi_e &= \text{internal strain energy} - \text{external energy due to applied loads} \\ &= \frac{1}{2} \int AE \left(\frac{\partial u}{\partial x} \right)^2 dx + \frac{1}{2} \int EI \left(\frac{\partial \theta}{\partial x} \right)^2 dx + \frac{1}{2} \int S \phi^2 dx + \frac{1}{2} \int GJ \left(\frac{\partial \gamma}{\partial x} \right)^2 dx \\ &\quad - \int q w dx - P w - C \theta\end{aligned}\quad (2.12)$$

The first four terms on the right hand side of equation (2.12) represent the axial, flexural, shear and torsional strain energies respectively. The last three terms represent, respectively, the work done by distributed loads, concentrated loads and concentrated couples.

The definition of the symbols involved in equation (2.12) is as follows:

AE = axial rigidity = cross-sectional area x elastic modulus,

EI = flexural rigidity = elastic modulus x 2nd moment of area about the relevant axis,

$$S = \text{shear rigidity} = \frac{GA}{\alpha}$$

$$= \frac{\text{shear modulus} \times \text{cross sectional area}}{\text{factor to allow for warping.}}$$

α will have in general two different values depending on the axis in which direction shear is acting. The value of α for some special cases is [5]

$$\alpha = \begin{cases} (12 + 11\nu)/10(1 + \nu) & \text{for a rectangular cross-section,} \\ (7 + 6\nu)/6(1 + \nu) & \text{for a circular cross-section,} \end{cases}$$

where ν is Poisson's ratio.

Typically [2] the value of α is 1.2,

GJ = torsional rigidity = shear modulus x torsional constant,

 $\partial u / \partial x$ = extension or axial strain, $\partial\theta/\partial x$ = pseudo-curvature or flexural strain,

ϕ = effective shear rotation or shear strain,

 $\partial\gamma/\partial x$ = torsional strain or angle of twist per unit length.

Note that all the integrals are taken over the length of the element.

2.3 Element Definition

The straight line non-prismatic element has three nodes - one at each end and the third somewhere in between as shown on figure 2.2.

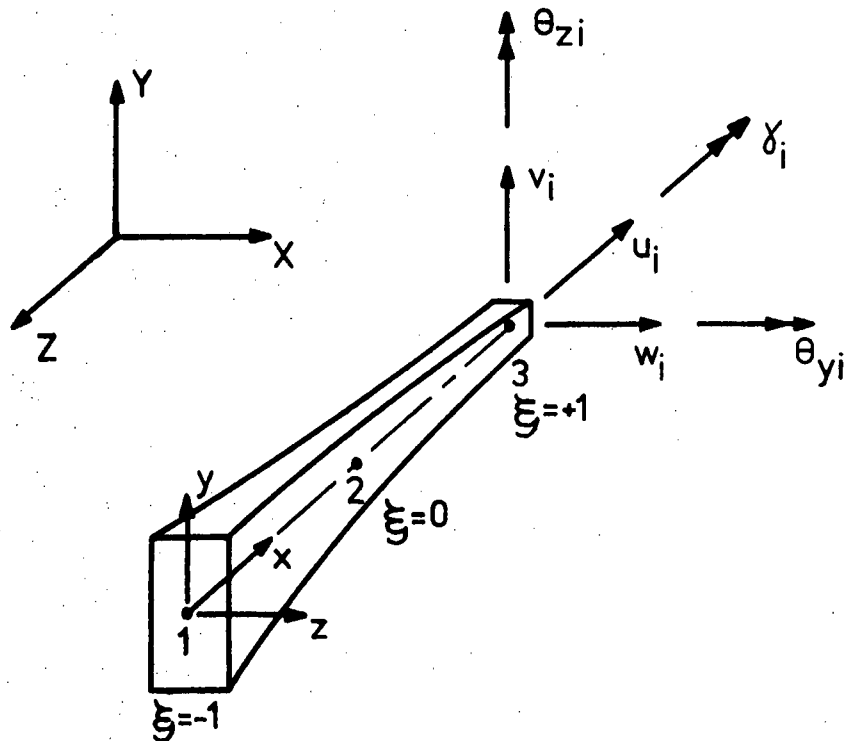


Figure 2.2 Non-prismatic parabolic isoparametric finite element definition.

Three coordinate systems are used in the element formulation:

- The natural system for the element where ξ ranges from - 1 to + 1 and is zero at the internal node [4]. It is stressed again that this internal node does not necessarily have to lie in the centre of the element.
- The element cartesian system (x, y, z).
- The system cartesian system (X, Y, Z).

Each node i has three translational and three rotational degrees of freedom and the element displacement vector can be written as

$$\delta^e = [\delta_1 \delta_2 \delta_3]^T = [u_1 \gamma_1 w_1 \theta_{y1} v_1 \theta_{z1} u_2 \gamma_2 w_2 \dots \theta_{y2} v_2 \theta_{z2} u_3 \gamma_3 w_3 \theta_{y3} v_3 \theta_{z3}]^T \quad (2.13)$$

A parabolic shape function is associated with each node as shown on figure 2.3 [2]:

$$N_1(\xi) = -\frac{1}{2}\xi(1 - \xi) \text{ associated with node 1,}$$

$$N_2(\xi) = (1 - \xi)(1 + \xi) \text{ associated with node 2,}$$

$$N_3(\xi) = \frac{1}{2}\xi(1 + \xi) \text{ associated with node 3.} \quad (2.14)$$

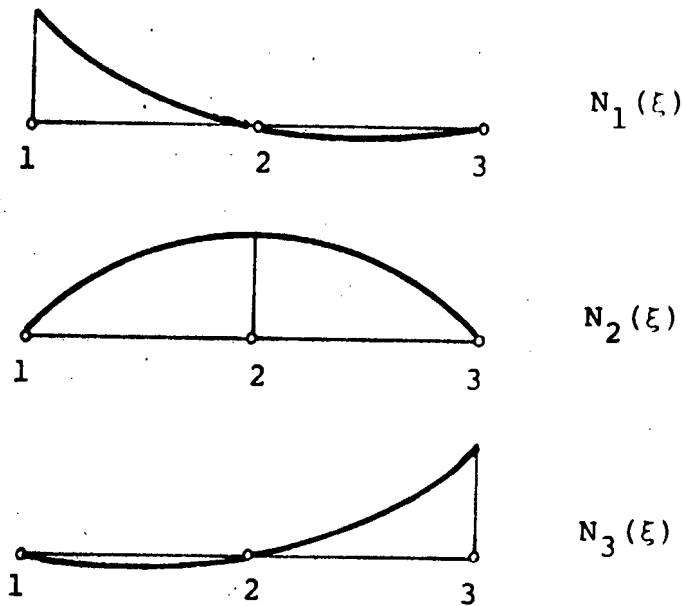


Figure 2.3 Element parabolic shape functions

The displacement field is then assumed to be expressed as

$$\delta(\xi) = N_1(\xi)\delta_1 + N_2(\xi)\delta_2 + N_3(\xi)\delta_3 \quad (2.15)$$

$$= \sum_{i=1}^3 N_i(\xi)\delta_i \quad (2.16)$$

Similarly, the geometry and corss-sectional property fields can be expressed in terms of the shape functions of equations (2.14) and the values of the particular quantity at the nodal points.

Thus,

$$x(\xi) = \sum_{i=1}^3 N_i(\xi) x_i \quad (2.17)$$

$$H(\xi) = \sum_{i=1}^3 N_i(\xi) H_i \quad (2.18)$$

where x and H refer to geometry and cross-sectional properties respectively.

2.4 Natural/Element Cartesian Coordinate Systems Relationship

This is achieved by the use of the Jacobian matrix, in this case, just an operator, defined as

$$J(\xi) = \frac{\partial x}{\partial \xi} = \sum_{i=1}^3 \frac{\partial N_i}{\partial \xi} x_i \quad (2.19)$$

This operator is required later in the formulation. For the moment it is clearly convenient to write J explicitly, i.e.

$$\begin{aligned} J(\xi) &= (-\frac{1}{2} + \xi)x_1 - 2\xi x_2 + (\frac{1}{2} + \xi)x_3 \\ &= \frac{x_3 - x_1}{2} + \xi(x_1 + x_3 - 2x_2) \end{aligned} \quad (2.20)$$

In the special case of the internal node being placed at the centre of the element, J becomes simply a constant scaling factor

$$J = L/2 \quad (2.21)$$

where L is the length of the element.

It will be seen later in the formulation that the inverse of J is needed and therefore this inverse has to exist. This inverse exists

provided that [4] there is a unique correspondence between the natural and element cartesian coordinates, as expressed by equation (2.17). In the present straight line element formulation this condition is clearly guaranteed.

2.5 Strain/Displacement Relationship

From equations (2.11) and (2.12) one can easily see how to express the element strain components in terms of the element nodal displacements. This relationship involves the shape functions and their derivatives as follows:

$$\epsilon = \sum_{i=1}^3 B_i(\xi) \delta_i \quad (2.22)$$

where B is the strain matrix and its i th component is expressed as

$$B_i(\xi) = \begin{bmatrix} \partial N_i / \partial x & 0 & 0 & 0 & 0 & 0 \\ 0 & \partial N_i / \partial x & 0 & 0 & 0 & 0 \\ 0 & 0 - \partial N_i / \partial x & N_i & 0 & 0 & 0 \\ 0 & 0 & 0 & \partial N_i / \partial x & 0 & 0 \\ 0 & 0 & 0 & 0 - \partial N_i / \partial x & N_i & 0 \\ 0 & 0 & 0 & 0 & 0 & \partial N_i / \partial x \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \partial u / \partial x \\ \partial v / \partial x \\ \phi_z = - \partial w / \partial x + \theta_y \\ \partial \theta_y / \partial x \\ \phi_y = - \partial v / \partial x + \theta_z \\ \partial \theta_z / \partial x \end{bmatrix}$$

(2.23)

(2.24)

and ϵ is the strain vector.

Note that the strain matrix B_i contains the shape function derivatives $\partial N_i / \partial x$ which are calculated from the chain rule expression

$$\partial N_i / \partial x = \frac{\partial N_i}{\partial \xi} \cdot \frac{\partial \xi}{\partial x} = \frac{\partial N_i}{\partial \xi} \cdot J(\xi)^{-1} \quad (2.25)$$

involving the inverse of the Jacobian operator defined previously.

2.6 Linear Elastic Stress/Strain Relationship

It is a major assumption that the cross-sectional properties are allowed to vary within the element and consequently the stress strain relationship is written as

$$\sigma = D(\xi)\epsilon = KH(\xi)\epsilon \quad (2.26)$$

where H is a matrix of cross-sectional properties and K is a constant matrix of material moduli.

D is generally called the stress matrix and given by

$$D(\xi) = \begin{bmatrix} EA(\xi) & & & & & \\ & GJ(\xi) & & & & \\ & & GA(\xi)/\alpha_z & & & \\ & & & EI_y(\xi) & & \\ & 0 & & & GA(\xi)/\alpha_y & \\ & & & & & EI_z(\xi) \end{bmatrix} \quad (2.27)$$

It should be noted that in the general non-prismatic case this matrix is variable along the element length, whereas in the simpler uniform member case the matrix becomes a constant within the element.

σ is the stress vector containing the force and moment resultants and given by

$$\sigma = \begin{bmatrix} N \\ T \\ Q_z \\ M_y \\ Q_y \\ M_z \end{bmatrix} \quad (2.28)$$

where N is the axial force along the element axis,
 T is the torsional moment,

Q_z is the shear force in local direction z,
 M_y is the bending moment about the element y axis,
 Q_y is the shear force in element direction y
 and M_z is the bending moment about the element z axis.

2.7 Element Stiffness Matrix Evaluation

Following the standard finite element procedure the element stiffness matrix can be calculated and the typical contribution associated with nodes i and j is given by the expression

$$K_{ij}^e = \int [B_i]^T D B_j dx \quad (2.29)$$

$$\text{where } dx = \det J(\xi) d\xi = J(\xi) d\xi \quad (2.30)$$

for this particular element.

Contributions K_{ij}^e are then evaluated using the well known Gauss-Legendre numerical integration techniques [1, 4, 8 - 10].

The integration techniques are expanded upon in Appendix B.

Thus, the element stiffness matrix is given by the following (18 x 18) symmetrical positive definite matrix

$$K^e = \begin{bmatrix} K_{11}^e & K_{12}^e & K_{13}^e \\ K_{21}^e & K_{22}^e & K_{23}^e \\ K_{31}^e & K_{32}^e & K_{33}^e \end{bmatrix} \quad (2.31)$$

The positive definiteness of the quadratic form of K^e is here accepted and proved in reference [11,12].

2.8 Space Transformations and System Stiffness Matrix

The element stiffness matrix thus obtained is referred to the local element coordinate system (x, y, z) and has to be transformed into the global system coordinates (X, Y, Z) as figure 2.4 indicates.

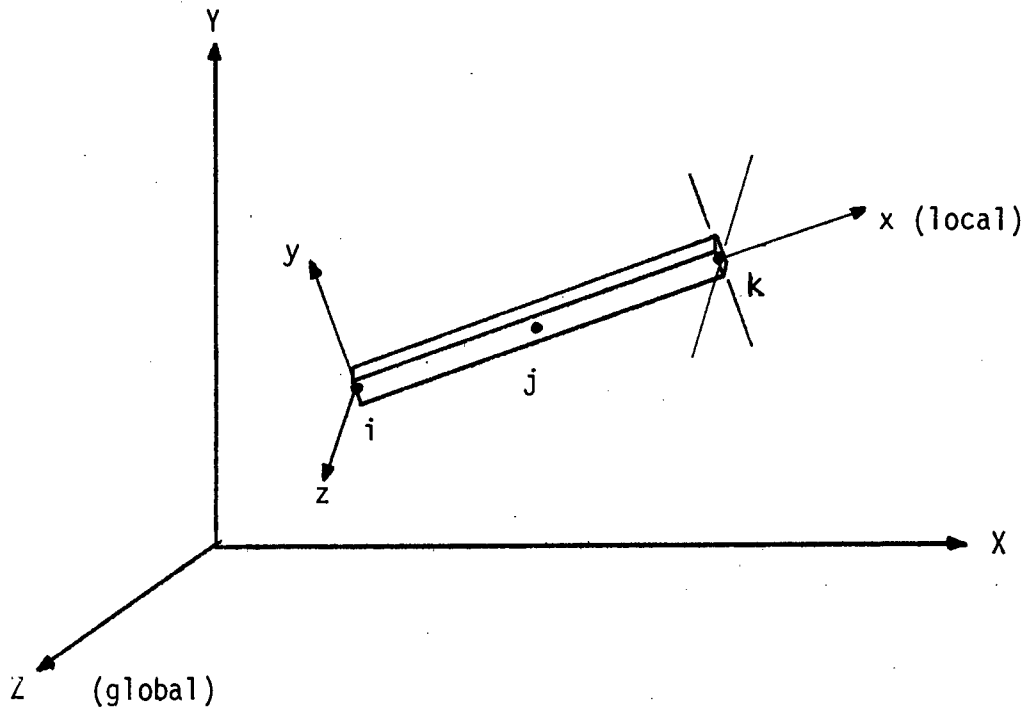


Figure 2.4 Element and system coordinate systems. Note that the local x axis is in the direction of increasing node numbering and that the local y,z coincide with the principal axes of the member cross-section.

To achieve this a standard axes transformation matrix T^e is compiled from the direction cosines of each particular element. A typical sub-matrix T_{ii}^e is expressed as [13]

$$T_{ii}^e = \begin{bmatrix} \cos(Xx) & 0 & \cos(Zx) & 0 & \cos(Yx) & 0 \\ 0 & \cos(Xx) & 0 & \cos(Yx) & 0 & \cos(Zx) \\ \cos(Xz) & 0 & \cos(Zz) & 0 & \cos(Yz) & 0 \\ 0 & \cos(Xy) & 0 & \cos(Yy) & 0 & \cos(Zy) \\ \cos(Xy) & 0 & \cos(Zy) & 0 & \cos(Yy) & 0 \\ 0 & \cos(Xz) & 0 & \cos(Yz) & 0 & \cos(Zz) \end{bmatrix}$$

(2.32)

where

$\cos(Xz)$ is the cosine of the angle between the element z coordinate

and the system X coordinate measured anticlockwise from the element to the system axes.

The automatic implementation of the axes transformations is done by the third node method of space transformations given in Appendix A.

Thus, the complete element transformation matrix (18 x 18) is symmetric and given by

$$T^e = \begin{bmatrix} T_{11}^e & 0 & 0 \\ 0 & T_{22}^e & 0 \\ 0 & 0 & T_{33}^e \end{bmatrix} \quad (2.33)$$

Finally, the element stiffness matrix in the global cartesian system is written as

$$K^S = [T^e]^T K^e T^e \quad (2.34)$$

2.9 Consistent Nodal Load Formulation

It is well known that in the finite element displacement method, the only permissible form of loading, other than initial stressing, is by the prescription of concentrated loads or couples at the nodal points. Consequently, concentrated loads or couples away from the nodes and any form of distributed loads or couples must be represented in terms of equivalent nodal loads F^S for the solution to proceed.

In any finite element program based on the isoparametric element principle, the calculation of equivalent nodal forces is not a process that can be performed manually since integrations over arbitrarily shaped regions are generally involved. Hence the equivalent nodal forces, due to pressures, etc., cannot be calculated for direct input and the inclusion of subroutines to perform this task becomes essential [2]. The integration techniques are discussed in Appendix B.

NONPRI has been written to allow for concentrated loads or couples, and linearly and parabolically varying loads or couples, thus covering all the possible loading conditions likely to occur in practice.

2.9.1 Point Concentrated Loads or Couples

To illustrate the principles involved, consider a point load P acting on a parabolic isoparametric beam element when the point of application does not coincide with a nodal point as shown on figure 2.5.

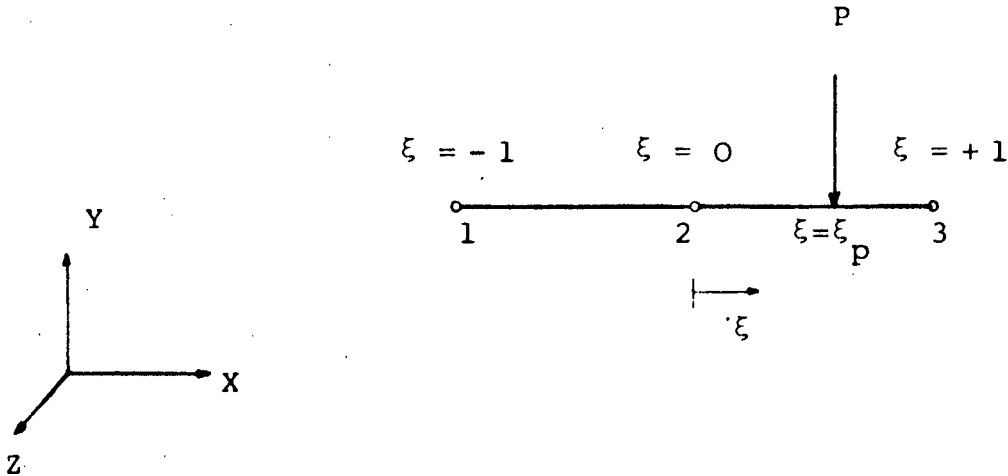


Figure 2.5 Application of off-node point load on one-dimensional parabolic isoparametric element.

Supposing the shape functions corresponding to the three nodal points are N_1 , N_2 and N_3 , let a virtual displacement w^* in the Y direction occur at each node in turn and let us denote the lateral nodal forces at each node which are equivalent to the applied load P as P_1 , P_2 and P_3 .

When the virtual displacement is applied to node i , by the principle of virtual work we have

$$P_i w^* = P N_i(\xi_p) w^* \quad (2.35)$$

where N_i is evaluated for the value of ξ at which the point load acts. Since this expression holds for an arbitrary displacement, w^* , then

$$P_i = P N_i(\xi_p) \quad (2.36)$$

This expression allows the equivalent nodal forces P_i to be determined by setting $i = 1$ to 3 in turn.

2.9.2 Distributed Loads or Couples

Let us consider, now, the case of a distributed load $q(\xi)$ /unit length as shown on figure 2.6.

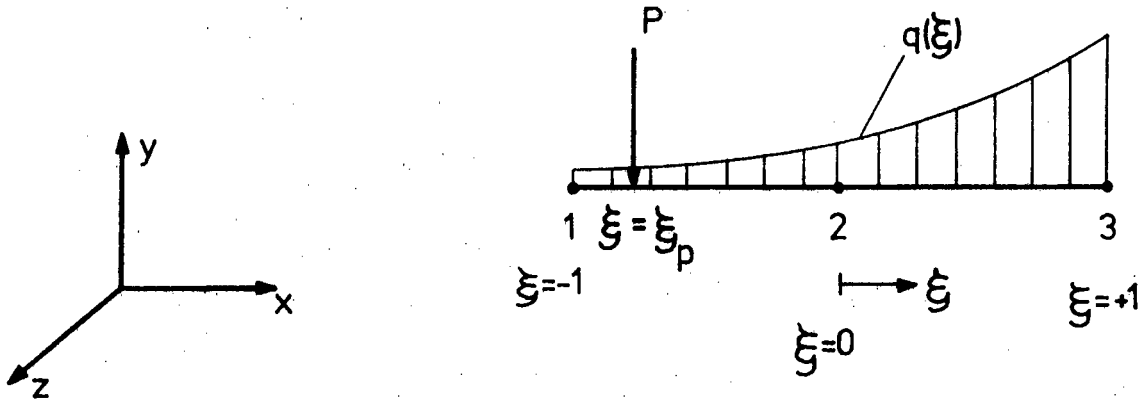


Figure 2.6 Application of distributed load on one-dimensional parabolic isoparametric element.

Again we make use of Virtual Work, and representing the applied nodal forces P_i and couples C_i by the vector

$$F^e = [P_{x1}, C_{x1}, P_{z1}, C_{y1}, P_{y1}, C_{z1}, P_{x2}, C_{x2}, P_{z2}, C_{y2}, P_{y2}, C_{z2}, P_{x3}, C_{x3}, P_{z3}, C_{y3}, P_{y3}, C_{z3}]^T \quad (2.37)$$

we can write for a distributed loading $q(\xi)$

$$\begin{aligned} P_i &= \int N_i q(\xi) dx \\ &= N_i q(\xi) \det J(\xi) d\xi \end{aligned} \quad (2.38)$$

$$C_i = 0 \quad (2.39)$$

The integral for P_i is calculated numerically using Gaussian integration. It should be noted that this particular element does not require any nodal couples in the discrete representation of a lateral distributed load or concentrated off-node point load. This is so because the lateral displacement w is independent of the rotation of the normal since $\theta = (\partial w / \partial x) + \phi$. (With the conventional 'cubic displacement' thin beam the lateral displacement w is not independent of the

rotation of the normal since $\theta = \partial w / \partial x$ and thus a lateral load produces lateral nodal forces and couples).

2.10 Force and Moment Resultants

Once the primary unknowns, the displacements in the case of the equilibrium equations, have been calculated from the solution of equation (2.10) one wishes to calculate the force and moment resultants in order to complete the solution of the problem. This is however not a straightforward procedure and the following considerations must be studied.

2.10.1 Optimal Sampling Points [1]

Figure 2.7 shows a curve which represents an assumed exact variation of the quantity (\bar{L}_u) and a set of piecewise linear least squares approximations to it (L_u) . It is clear that at some points within each segment the approximate solution must coincide with the exact one. If one knew in advance where such points occur one could always find the exact solution of these points.

A useful property of Gauss-Legendre numerical integration is of great importance here. This property states: 'If one devises a numerical integration formula with a minimum number of sampling points which just integrates precisely a polynomial of degree $2M + 1$, then in general at such points a polynomial of order $M + 1$ is equal to its least squares approximation by a polynomial of order M '.

The above statement is exactly true for one-dimensional Gauss point integration.

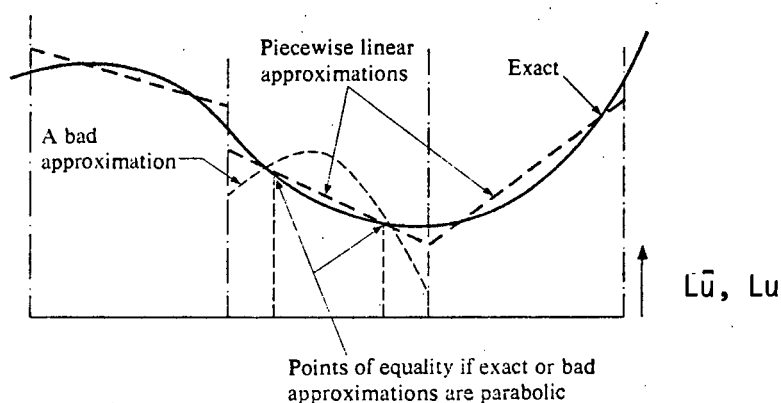


Figure 2.7 Piecewise linear least squares approximation to the exact curve (\bar{L}_u) .

It is clear from the example presented that if the exact curve was a parabola, then two Gauss points would define uniquely a straight line which happens to be the least squares approximation to it. Conversely, if one sampled the approximation Lu at these points, one would then obtain an accuracy of one order greater than that available elsewhere by the approximation. Obviously, such points are optimal for sampling the quantity Lu , or in more realistic terms the stresses in an elasticity problem. Therefore [1], at numerical integration points which just integrate exactly a polynomial of order $2(p - m) + 1$, i.e. with an error of order $O(h^{2(p-m)+2})$, the approximation to Lu will be nearly one order better, i.e. $O(h^{p-m+2})$, where p is the order of the approximation and m is the differentiation order implied in the operator L . In any finite element computation it pays, therefore, to sample the stresses at integration points and this has been realized by many researchers [2, 15 - 17].

Figure 2.8 shows the analysis of a cantilever beam using four quadratic 'serendipity' elements. Although the results for deflections are excellent, the shear stresses show a parabolic variation in each element which gives a very poor representation of the true stresses at the nodal points [1]. On the other hand, the values sampled at the Gauss points are very good.

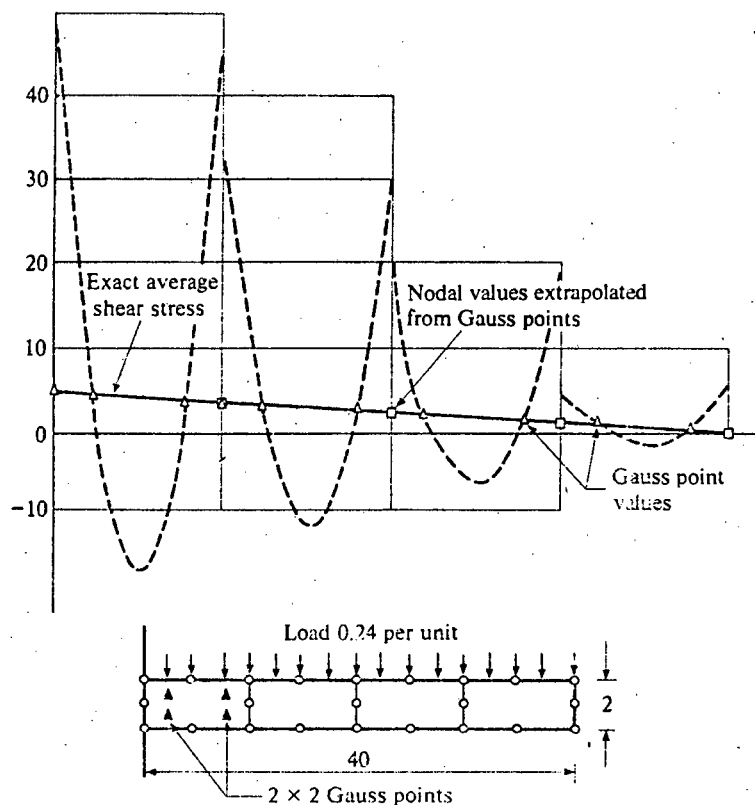


Figure 2.8

Cantilever beam with four parabolic elements. Stress sampling at Gauss points and linear extrapolation at nodes.

The cantilever example suggests that in quadratic elements the stresses should never be calculated at the nodes. Some methods of extrapolating the values of stresses at the nodes have been devised [2, 18] but these are not always satisfactory, especially in the case of beam elements. Therefore other ways of calculating the force and moment resultants at the nodes have to be found and these are presented in sections 2.10.3 and 2.10.4. However, the force and moment resultants at the Gauss points are also useful as will be shown in section 2.11 and this calculation is dealt with now.

2.10.2 Sampling Resultants at Integration Points

This is a simple procedure and is, in fact, equivalent to calculating internal forces at the integration points [14]. The expression used is given by

$$\sigma(\xi_{GP}) = D(\xi_{GP}) B(\xi_{GP}) \delta^e \quad (2.40)$$

for the resultant values in the system set of axes. One, however, usually requires the resultants for each particular element in its local set of axes and (2.40) is then modified to

$$\sigma(\xi_{GP}) = D(\xi_{GP}) B(\xi_{GP}) T^e \delta^s \quad (2.41)$$

where, notice, D and B are evaluated at the Gauss Points

and δ^s is the vector of global displacements.

T^e is the element transformation matrix.

2.10.3 Sampling Resultants at External Nodes

As stated in section 2.10.1 a different approach has to be pursued in order to evaluate the resultants at the external nodes [5, 14]. This is equivalent to calculating external forces at the ends of the element. This procedure is based on a theorem due to Castigliano which states that the differential of the strain energy with respect to a displacement gives the force in the direction of that displacement.

The element strain energy is given as

$$SE = \frac{1}{2} [\delta^S]^T K^e \delta^S - [\delta^S]^T F^S \quad (2.42)$$

$$\text{and, } \frac{\partial (SE)}{\partial \delta^S} = K^e \delta^S - F^S \quad (2.43)$$

Thus, the complete expression for the force and moment resultants at the end nodal points in the element cartesian axes system, is written as

$$F^e = K^e T^e \delta^S - [T^e]^T F^S \quad (2.44)$$

However, the force and moment resultants at the internal node still remain to be calculated.

2.10.4 Sampling Resultants at Internal Node

This is the one problem for which there is no direct method of solution within the ideas of the finite element method. Therefore a method which can perhaps be considered 'untidy' within the finite element argument is adopted. Simple equilibrium equations (statics) are employed to find the resultants at the internal node starting from those at one of the external nodes.

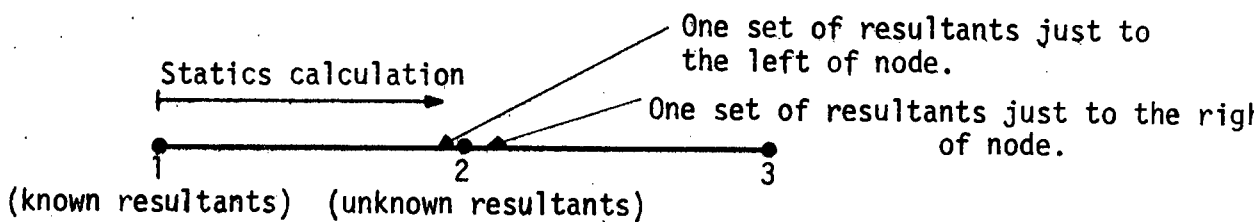


Figure 2.9 Direction of statics calculation for obtaining force and moment resultants at the internal node.

Figure 2.9 shows more specifically the operations that are carried out:

- (i) Start with known resultants at node 1.

- (ii) Include the contribution, if any, of external loads applied between nodes 1 and 2. This yields a set of resultants for node 2 corresponding to a point just to the left of 2.
- (iii) Include the contribution, if any, of external loads applied at node 2. This yields a second set of resultants for node 2 corresponding to a point just to the right of 2.

Note that any discontinuity at node 2 in the functions representing the force and moment resultants is exactly modelled in this way.

This completes the force and moment resultant calculations of the present formulation.

2.11 Interpolation Facilities

Having now calculated force and moment resultants at six different positions along the element, as shown on figure 2.10, it is possible to accurately interpolate values of these quantities anywhere along the element.

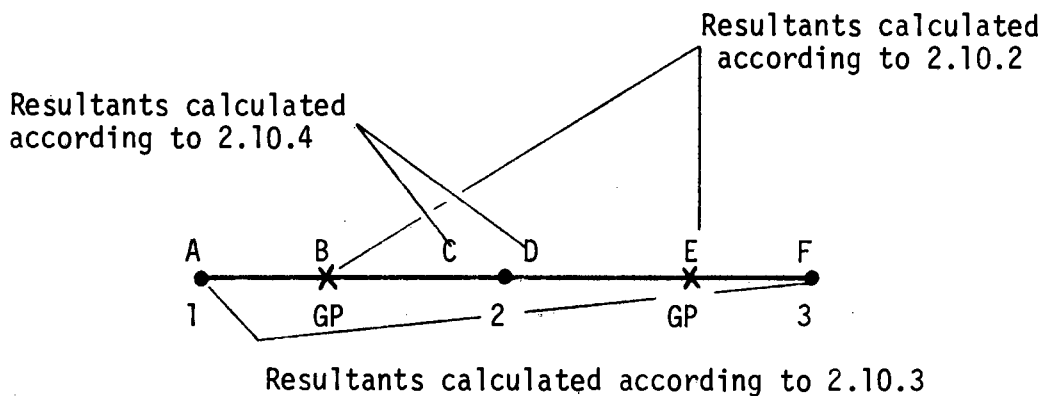


Figure 2.10 Sampling positions of force and moment resultants within each element.

Parabolas are fitted on each 'half-element' through the 3 sampled positions, i.e. A, B, C for the left hand side 'half-element' and

D, E, F for the right hand side 'half-element'. This fitting through unevenly spaced points is done by using Lagrangian interpolation polynomials which are found to be very accurate. The topic of Lagrangian interpolation polynomials is expanded upon in section 2.13. Of course, it should be realized that saying parabolas are fitted, means that straight lines can also be fitted as a special case of a parabola.

Thus, a large amount of extra information can be obtained at minimal computational effort. This has important implications as far as the finite element modelling using the present formulation is concerned. The subject of modelling within the context of this formulation is dealt with in section 2.15.

2.12 The Non-Prismatic Feature

As stated in section 2.6, in the general case of a non-prismatic element, the stress matrix D is variable along the element length, meaning that when evaluating the stiffness of the element using equation (2.29), it is necessary to interpolate the values of the cross-sectional properties of D at the Gauss integration points. This is done, as mentioned before in equation (2.18), by using the shape functions of equation (2.14), i.e.

$$H(\xi) = \sum_{i=1}^3 N_i(\xi) H_i \quad (2.18)$$

It obviously requires the input of the cross-sectional properties H at the three element nodes.

2.13 The Element with Unevenly Spaced Nodes and Use of Lagrangian Interpolation Polynomials

The need and acquired advantages of being able to arbitrarily define the geometrical position of the internal node within the limits of the element, of course, was motivated in Chapter 1. Let us now examine the implications brought about by this extra variable in the formulation. Consider, as an example, the element shown on figure 2.11 defined with its internal node three tenths of the element length

away from external node 1.

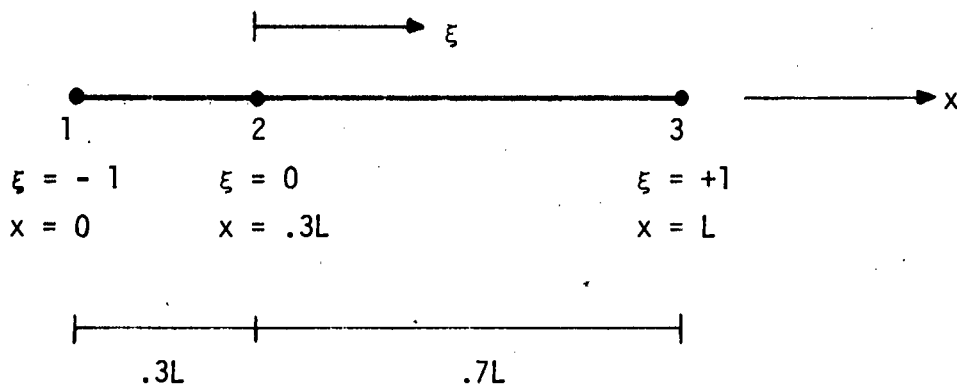


Figure 2.11 Element definition in the case of unequally spaced nodes.

Note that the origin of the natural coordinate ξ is still the same even though the internal node is not in the centre of the element. Also the limiting natural coordinates corresponding to nodes 1 and 3 are still -1 and $+1$ respectively. In terms of the element cartesian coordinates the condition depicted in figure 2.11 can be visualised as compressing the element between nodes 1 and 2 and stretching it between nodes 2 and 3.

The shape functions given in equations (2.14) remain unchanged since they are in terms of the natural coordinate ξ which is also unchanged.

The significant difference comes in the fact that the Jacobian operator of equation (2.19) is now a variable from Gauss point to Gauss point. Rewriting equation (2.20)

$$J(\xi, x_i) = \frac{x_3 - x_1}{2} + \xi(x_1 + x_3 - 2x_2) \quad (2.20)$$

one realizes that J is dependent on the nodal cartesian coordinates as well as on the natural coordinate. This implies that the $\det J(\xi, x_i)$ used in the integration procedures to obtain the element stiffness matrix and the equivalent nodal loads also becomes a variable and in

this way the uneven spacing of the element nodes is accounted for. Lastly, a point to stress is that when interpolating quantities at the natural level no modification in the present formulation (i.e. shape functions can still be used) is necessary since the natural coordinates are unchanged by the arbitrary definition of the internal node. However, when interpolating quantities at the cartesian coordinate level, the shape functions of (2.14) are no longer valid and a new method of interpolating from unevenly spaced points is called for. The method used is interpolation through Lagrangian Interpolation Polynomials which are specifically designed for this purpose. Before getting on to these polynomials it is perhaps useful to state that the condition of interpolation at cartesian coordinate level arises in important formulation steps such as the calculation of cross-sectional properties and distributed load intensities at the Gauss integration points. It arises again when interpolating force and moment results anywhere along the element as mentioned in section 2.11.

The Lagrangian Interpolation Polynomial is a development which is parallel to interpolation using the shape functions [19], but the spacing between the values used for interpolation need not be equal nor need the values used for interpolation be in any specific order. In the present case, with three values for interpolation available, a parabolic is the highest degree polynomial possible.

Letting the cartesian coordinates of the three nodes be x_1 , x_2 and x_3 , and letting the values of the quantity under consideration at the three nodes be f_1 , f_2 , f_3 respectively, one can write the Lagrangian form as

$$P_2(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} \cdot f_1 + \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} \cdot f_2 + \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)} \cdot f_3 \quad (2.45)$$

where x represents the cartesian coordinate of the point where the quantity under consideration is to be interpolated and $P_2(x)$ is the parabolic Lagrangian polynomial.

Note that each term is parabolic and therefore $P_2(x)$ is also parabolic.

The error term for this kind of interpolation polynomial is of comparable accuracy to the error term of the shape function approximation [19].

2.14 Degenerate Forms of the Element

In Chapter 1, the motivation put forward for formulating degenerate forms of the six degree of freedom per node 'parent' element was the considerable advantages of the acquired modelling flexibility of the element in practical analysis problems. Two degenerate forms are formulated in this work namely, a pure truss element and a truss/frame transition element. The applicability of the pure truss element in an assemblage of truss elements exclusively, is quite obvious. However, the applicability of the pure truss and truss/frame transition forms when used in conjunction with the pure frame element is perhaps not as apparent.

Consider the example of the simple bridge structure sketched on figure 2.12.

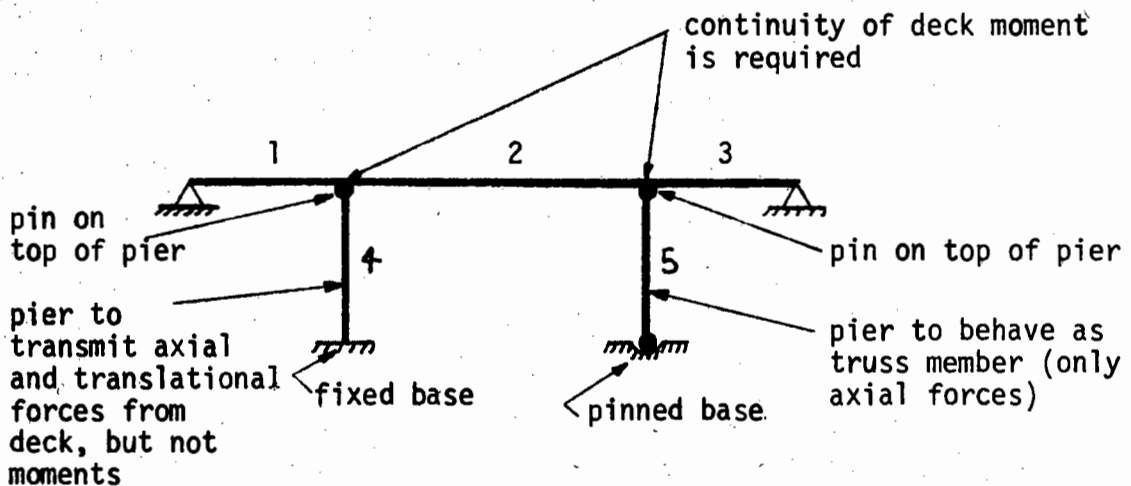


Figure 2.12 Application of pure truss and truss/frame transition degenerate forms of the element.

The practical situation depicted on figure 2.12 requires considerable care as far as modelling of the structure is concerned. Considering the deck (composed by members 1, 2 and 3) one wants continuity of bending moment throughout, i.e. no deck moments must be transmitted down to the piers and the deck must simply behave as a three span continuous beam. This can be achieved by modelling elements 1, 2 and 3 as pure frame elements, i.e. in the most general three dimensional case, with six degrees of freedom per node. However, this is not possible if the piers are not modelled in a compatible manner. Pier element 5 is required to behave as a pure truss member, i.e. only axial forces will be catered for. Element 5 is then modelled as a pure truss element, which, in the three dimensional case, has three global translational degrees of freedom but they all result in a single axial effect along the element - this is equivalent to pinning element 5 top and bottom. Lastly, pier element 4 is required not to transmit any moments from the deck, i.e. truss node on top of the pier, but for the rest of the way down should behave as a normal frame element. This requires the modelling of the top node of 4 as a three translational degrees of freedom node, and the modelling of the two remaining nodes as six degrees of freedom nodes in the most general three dimensional case.

Let us now examine how the degenerate forms of the element are formulated.

2.14.1 The Truss/Frame Transition Element

In this form one requires the three degrees of freedom associated with the axial force and the two shear forces.

Therefore, at the transition node one can rewrite the strain energy expression as

$$\Pi = \frac{1}{2} \int AE \left(\frac{\partial u}{\partial x} \right)^2 dx + \frac{1}{2} \int S \phi^2 dx \quad (2.46)$$

involving the axial and shear contributions only.

The strain/displacement relationship at the transition node i is then written as

$$\begin{bmatrix} \partial u / \partial x \\ - \\ \phi_z \\ - \\ \phi_y \\ - \end{bmatrix} = \begin{bmatrix} \partial N_i / \partial x & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\partial N_i / \partial x & N_i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\partial N_i / \partial x & N_i \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_i \\ \gamma \\ w_i \\ \theta_{yi} \\ v_i \\ \theta_{zi} \end{bmatrix} \quad (2.47)$$

where the elements corresponding to the rotational degrees of freedom have been eliminated.

This transition formulation will obviously only yield resultant forces and no resultant moments at the transition nodes.

2.14.2 The Pure Truss Element

This highly degenerate form considers only the node degree of freedom associated with axial force. Thus, the strain energy expression is further reduced

$$\Pi = \frac{1}{2} \int AE \left(\frac{\partial u}{\partial x} \right)^2 dx \quad (2.48)$$

to involve the axial strain contribution alone.

The strain/displacement relationship becomes that of a pure truss and can be written as

$$\begin{bmatrix} \partial u / \partial x \\ - \\ - \\ - \\ - \\ - \end{bmatrix} = \begin{bmatrix} \partial N_i / \partial x & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_i \\ \gamma_i \\ w_i \\ \theta_{yi} \\ v_i \\ \theta_{zi} \end{bmatrix} \quad (2.49)$$

where all the elements have been deleted except for the one relating to axial effects.

Clearly, this degenerate form formulation only yields one resultant axial force.

It should be noted that in the pure truss element only loads at the end nodes are allowed, i.e. no forms of distributed loads along the element can be specified.

2.15 Modelling of Structures using the NONPRI Concept

The non-prismatic and interpolation capabilities of this formulation clearly lead the analyst to think in terms of modelling using the least number of elements per member or span. Certain factors have to be carefully considered when modelling, namely, discontinuity of distributed loading functions within a member and marked discontinuities in the cross-sectional profile within a member. But, even for some discontinuities, approximations can be made which yield good results, as will be shown in this work.

Some examples of modelling using NONPRI are now presented.

- Modelling of prismatic continuous beam with various load types

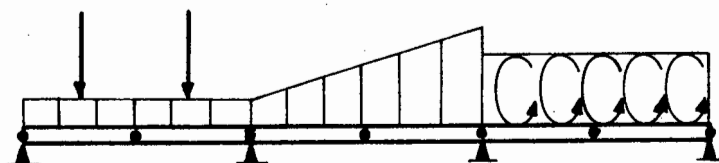


Figure 2.13 Continuous beam with various load types

Three elements can be used in this problem to get a complete solution - one per span.

- Modelling of two-bay portal frame with short haunches.

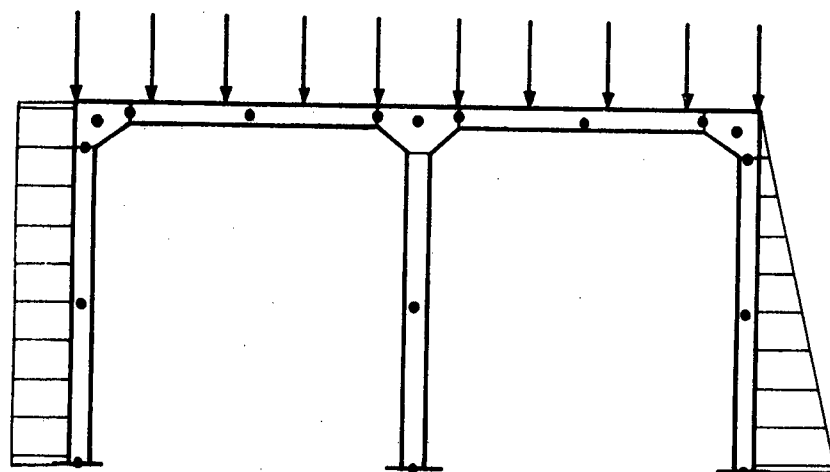


Figure 2.14 Two-bay portal frame with short haunches.

Eight elements can be used for a complete solution - one for each member and one for each haunch.

- Modelling of the same two-bay portal frame but with long haunches and travelling crane support.

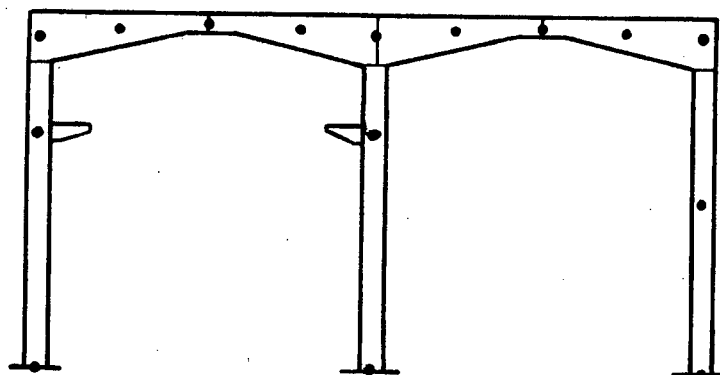


Figure 2.15 Two-bay portal frame with long haunched beams and travelling crane support.

In this case, each of the portal beams is approximated by two elements with a linearly varying cross-section profile. The columns supporting the travelling crane are modelled by elements having an off-centre internal node. Therefore seven elements are necessary for a complete solution.

- Modelling of non-prismatic bridge structure.

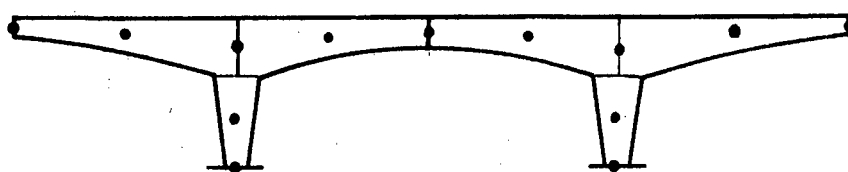


Figure 2.16 Bridge frame with non-prismatic members

Provided there are no distributed load function discontinuities within an element, the structure can be modelled with six elements as shown on figure 2.16.

- Modelling of a simple beam with a triangular load.

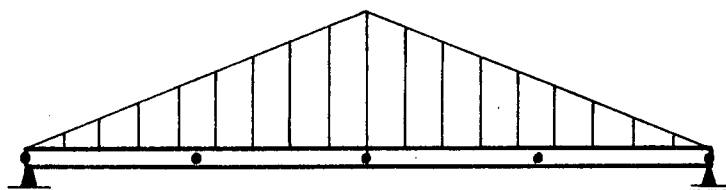


Figure 2.17 Simple beam with triangular load.

Two elements would be needed to model this span due to the distributed loading function discontinuity.

- Modelling of multi-storey frame

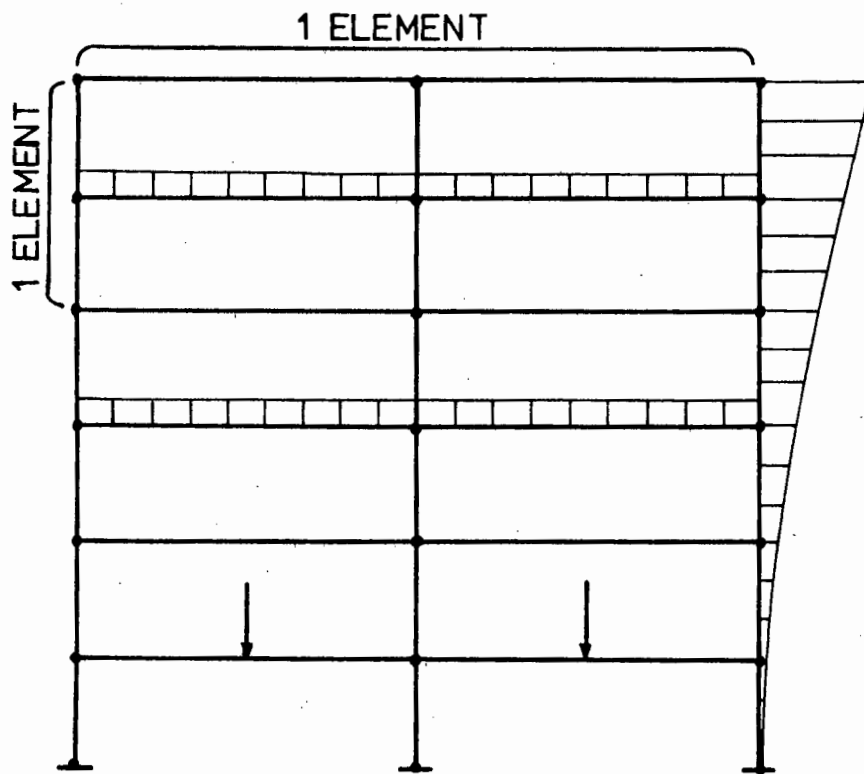


Figure 2.18 Multi-storey frame

In this case one could go as far as modelling each two adjacent members using one single element.

In conclusion, it can be said that the finite element modelling of structures using NONPRI depends mainly on the nature of the loads and/or the variation of cross-sectional profiles, thus the analyst's skill can be well employed in this sector.

CHAPTER 3

SOLUTION OF EQUATIONS OF STATIC ANALYSIS - A FRONTAL APPROACH

3.1 Introduction

It can be said that the basic steps for deriving a finite element solution (by the displacement method) to an equilibrium problem can be summarised as :

- i) Sub-division of the structure into finite elements
- ii) Evaluation of the element stiffness terms
- iii) Evaluation of the element load terms
- iv) Assembly of the element stiffness terms into the overall stiffness matrix
- v) Assembly of the element load terms into the overall load vector
- vi) Solution of the resulting linear simultaneous equations for the primary unknown variables (nodal displacements)
- vii) Evaluation of subsidiary element quantities such as stresses, forces and moments.

Parts (i), (ii), (iii) and (vii) have been dealt with in Chapter 2. We set out, in Chapter 3, to discuss the remaining operations, i.e. assembly and solution.

3.2 Choice of Solution Technique

The overall efficiency of an analysis program depends to a large extent on the numerical techniques used for the solution of the system equilibrium equations. This is much more so in the case of the solution of the equations of dynamic analysis. For this reason a lot of research has been done in this field [4].

Basically, there are two different types of methods for the solution of simultaneous equations : direct solution techniques and iterative solution techniques. In a direct solution the equations of (2.10) are solved using a predetermined sequence of operations in an exact manner, whereas an iterative procedure has to be used when an iterative method of solution is considered. Currently, in almost all applications the direct methods are the most effective. Within the scope of the direct methods several procedures exist and

Irons [21] , Bathe and Wilson [4] and Mondkar and Powell [24] are only some of the researchers of note in the field.

In NONPRI, the frontal method of equation assembly and reduction originated by Irons [21] has been adopted for its effectiveness, small storage requirements and possibility of solving very large systems.

3.3 General Description of the Frontal Solution Technique

This method is already well known and tested and it has proved to be a very efficient one. The actual routine on which the NONPRI solution is based is the one presented by Hinton and Owen [2] and modified in certain routine operations. Precisely for this reason, I will not attempt to give a detailed discussion of the solution routine in this Chapter and rather present the main features of the frontal method. Hinton and Owen [2] and Cheung and Yeo [22] can be consulted for a more detailed explanation of the method.

The frontal solution used in NONPRI is capable only of the solution of symmetric systems of linear stiffness equations.

One could say that the principal idea of the frontal solution is to assemble the equations and eliminate the variables simultaneously. As soon as the coefficients of an equation are completely assembled from the contributions of all relevant elements, the corresponding variable is eliminated. Therefore the complete stiffness matrix is never formed at any instant, since immediately after elimination the reduced equation is transferred to disc storage. The core holds, at any instant, the upper triangular part of a square matrix containing the equations being formed at that time. These equations, their corresponding nodes and degrees of freedom are called front. The number of unknowns in the front is called the frontwidth, and this changes continually during the element assembly/equation reduction process. The maximum size of the problem which can be solved is dictated by the maximum frontwidth. The equations, nodes and degrees of freedom

in the front are called active, those yet to be considered are inactive, those already eliminated are said to be deactivated.

3.4 Prefrontal Routine and Suggested Modification

Since each equation to be assembled and eliminated corresponds to a particular degree of freedom rather than a node number it becomes necessary to rearrange the data which relates to prescribed displacements at nodes. The fixity code and prescribed displacement values are transferred to vector arrays spanning the total number of degrees of freedom in the structure. Index IPOSN determines the position of a particular nodal degree of freedom in the vector. $IPOSN = (IPOIN-1)*NDOFN + IDOFN$ is the position of the IDOFNth degree of freedom of node IPOIN.

In order to know when a nodal variable (degree of freedom) can be eliminated it is necessary that the last appearance of a node in the element nodal point listings be recorded. To register this final appearance a negative sign is placed in the LNODS (i.e. topology) entry before the corresponding nodal identification number IPOIN. This is done for each nodal point number in turn, by Hinton and Owen [2] and Irons [21], as follows :

C

C***CHANGE THE SIGN OF THE LAST APPEARANCE OF EACH NODE

C

```

      DO 140 IPOIN=1, NPOIN
      KLAST=0
      DO 130 IELEM=1, NELEM
      DO 120 INODE=1, NNODE
      IF(LNODS (IELEM,INODE).NE.IPOIN) GO TO 120
      KLAST=IELEM
      NLAST=INODE
120    CONTINUE
130    CONTINUE
      IF(KLAST.NE.0) LNODS (KLAST,NLAST)=-IPOIN
140    CONTINUE
```

where

NPOIN = number of structural nodes
 NELEM = number of elements
 NNODE = number of nodes per element
 LNODS = element topology.

Note that the inner loop of the above coding is obeyed $NPOIN \times NELEM \times NNODE$ times.

An alternative strategy is suggested by Collins [23]. It involves the use of an integer vector NFOUND of length NPOIN, as follows :

```

C
C***CHANGE THE SIGN OF THE LAST APPEARANCE OF EACH NODE
C
      DO 120 IPOIN=1, NPOIN
120    NFOUND(IPOIN)=0
      DO 140 IELEM=1, NELEM
      IA=NELEM+1-IELEM
      DO 130 INODE=1, NNODE
      JA=NNODE+1-INODE
      NODENO=LNODS(IA,JA)
      IF(NFOUND(NODENO).NE.0) GO TO 130
      NFOUND(NODENO)=1
      LNODS(IA,JA)=-NODENO
130    CONTINUE
140    CONTINUE

```

where NODENO = the node number.

Note, now that the inner loop of this alternative strategy is only obeyed $NELEM \times NNODE$ times giving therefore a significant reduction in computational effort.

3.5 Formulation of Housekeeping Arrays

As mentioned previously the frontal technique operates by performing simultaneously assembly and elimination. By eliminating variables

(degrees of freedom) as soon as their assembly is complete, core storage is made available for nodal variables not yet assembled. To keep this procedure going a quite elaborate housekeeping system is necessary, in particular the recording of the position in the front into which each degree of freedom of an element is to be assembled as well as recording of a list of the active variables currently in the front.

The assembly position in the front of each degree of freedom of a given element is controlled by the use of the array NDEST(KDOFE), where KDOFE ranges over the number of degrees of freedom per element. Since it defines the destination of the variable NDEST is called the destination vector. It is a local array and is computed separately on the introduction of each element.

Each element nodal variable is first assigned a location according to the algorithm $LOCEL(NPOSI) = (LNODS(IELEM, INODE) - 1) * NDOFN + IDO FN$

The right hand side of the expression states that the IDO FNth degree of freedom of the INODEth node of a particular element IELEM is given a location increasing with nodal point numbering and with increasing number of degrees of freedom. The left hand side indicates that this information is stored in the vector LOCEL - variable location vector - with position in the vector being defined by $NPOSI = (INODE - 1) * NDOFN + IDO FN$ where INODE defines the element nodal point and IDO FN defines the nodal degree of freedom.

Array LOCEL is constructed for each element in turn before the assembly/elimination process begins. If, when dealing with a particular element, it is found that the variable is making a final appearance then a negative sign is also placed before the corresponding entry in the LOCEL vector.

The vector of active variables contains a list of all the variables currently in the front and is called NACVA(IFRON), where IFRON spans over the maximum permissible length of the front, MFRON. This states that the IFRONth equation of the front relates to variable NACVA(IFRON)

where the variable location is as defined by LOCEL. A zero value in NACVA indicates an available space which can be employed in the assembly of later elements. The content of NACVA is changing constantly as the front moves through the elements of the structure.

3.6 Assembly and Elimination Process

For each element assembly proceeds in the same way as in the direct stiffness approach. The element stiffness matrix, SSTIF, and loads associated with the element, ELOAD, are introduced from disc file and core storage respectively. Only one half of the element stiffness matrix SSTIF is ever considered and thus the reason to state that this routine is restricted to symmetric matrices. By means of the destination vector NDEST the global row and column of any element entry are found and therefore it is possible to sum that element stiffness contribution SSTIF to a degree of freedom (variable) global stiffness GSTIF. The same goes for summing the element loads ELOAD to the global loads GLOAD. To improve program efficiency, since only half of the stiffness matrix is considered, GSTIF 2-D array is transformed to a 1-D array. This is achieved by the functions transforming the matrix form $A(I,J)$ to the vector form $A(N)$

$$N = \text{NFUNC}(I,J) = \frac{I(I-1)}{2} + J \text{ if } I \geq J \text{ (store lower triangle)}$$

or

$$N = \text{NFUNC}(I,J) = \frac{J(J-1)}{2} + I \text{ if } I \leq J \text{ (store upper triangle)}$$

Both functions give the same final vector starting from a symmetric matrix.

In the ordinary Gaussain process the variables are eliminated in the order in which they are encountered going down the matrix. However, in the frontal technique the order of elimination is different from the order of formation of the equations. Also, the order of formation is not straight forward but is controlled by the available space in the front.

When the elimination reaches a prescribed variable, the procedure becomes trivial. The right hand sides of the system of equations are modified and the corresponding column of the front matrix, except the diagonal term, is set to zero.

At the end of the assembly/elimination procedure, a disc file will have been created with a total number of records equal to the total number of degrees of freedom in the structure. Each record corresponds to an eliminated variable equation, the order of equations being as governed by the front which is in turn governed by the order of element numbering.

A record is kept of the order of elimination using the variables IFRON and NIKNO. Therefore for each eliminated equation - 1 record - we require the following information to be transferred to disc file.

EQUAT(MFRON) - the reduced frontal equations

EQRHS - the reduced right hand side loading terms

IFRON - the position of eliminated variable in the front at the time of elimination

NIKNO - the variable identification name.

3.7 Back Substitution Phase

The back substitution process can be said to be a frontal process in reverse. The disc file has been left in its final position after assembly/elimination and information relating to the last equation to be eliminated can be obtained by BACKSPACING the disc file by one record. Therefore, one keeps recalling information from disc file by backspacing one record at a time and back substituting from the last equation upwards until all displacements have been calculated. The displacements are stored in a global vector of nodal displacements called ASDIS. Although the elimination takes place in an unconventional manner, the reduced matrix which is transferred to disc file can be finally treated in the usual way for back substitution. Moving upwards from the last equation, each new equation considered introduces only one unknown quantity which can be directly calculated.

In cases where the nodal displacements are prescribed (boundary conditions) or prescribed displacements the nodal reactions are also found.

3.8 Remarks on the Frontal Solution Resolution Facility

For an economic processing of multiple loading cases (very relevant for design purposes) an equation resolution facility becomes essential. For this, the reduced equations are stored in their eliminated form and subsequent solutions simply involve the reduction of the right hand side load terms. Therefore, savings are made in the element stiffnesses formulation and in the element stiffness assembly and reduction which need not to be repeated. The multiple load cases, however, involve the use of new disc files for the storage of loading terms other than the ones referring to the first load case.

3.9 Frontal Solution Operation

The frontal solution routine used in this program is a self-contained solver which could be employed in any finite element program. It assumes that each nodal point has the same number of degrees of freedom, this number being however optional. It also assumes that the element stiffness matrices and load vectors have been previously generated in the program and are stored and available from disc file. The frontal solution subroutine is very long and involved, its main operations being the ones described by the flowchart now presented [2] on Figure 3.1.

3.10 Advantages of the Frontal Solution Method in Static Analysis

There are many advantages in the frontal solution method and as a result it has been accepted as a powerful and efficient equation solver.

Nodal numbering in the frontal method is irrelevant while element numbering is crucial. This is immediately an advantage in isoparametric elements since the number of elements is always far less than the number

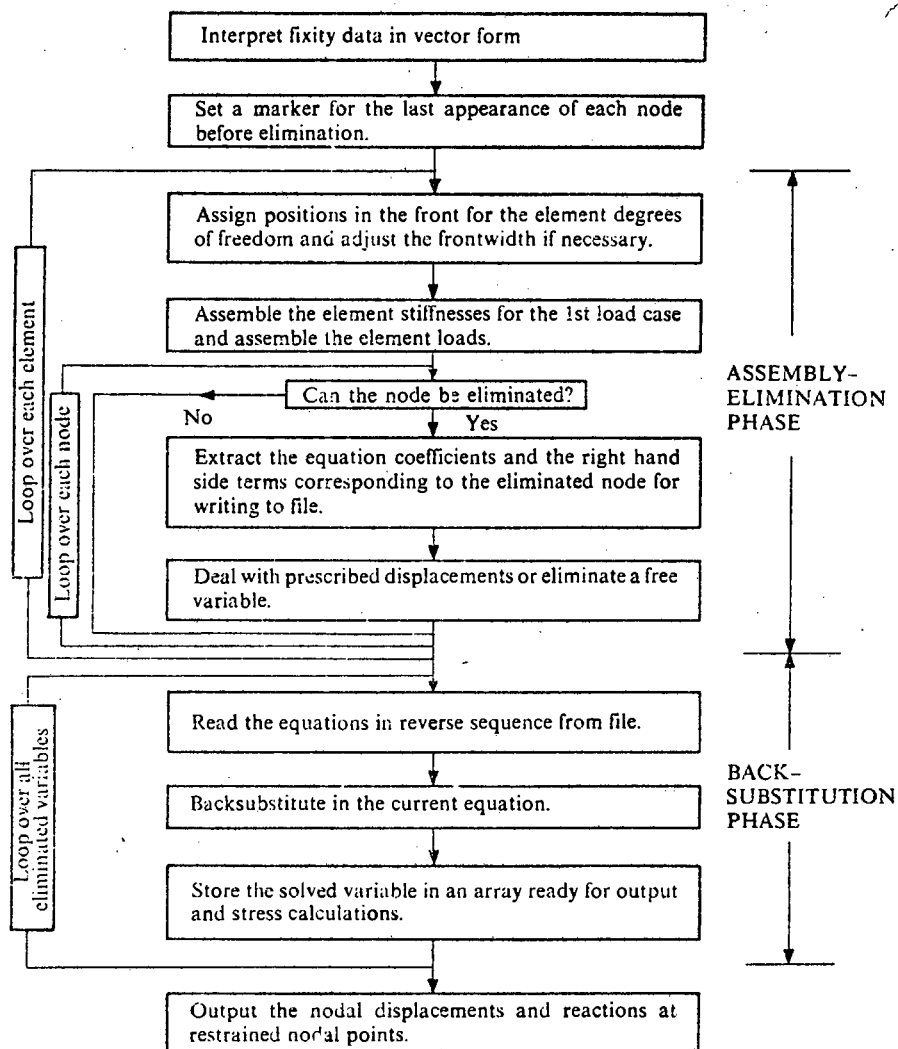


Figure 3.1

Main operations of frontal solution.
(from Hinton and Owen [2]).

of nodes (especially in 3-dimensional isoparametrics). Variables, in the frontal method, are introduced at a later stage and eliminated earlier than in a conventional band solver, i.e. the variables active life is shorter. This helps reduce numerical inaccuracies due to roundoff error.

The frontal method can be called an 'out of core' solver since it requires in core a minimum storage which is at the very most (and very seldom) the same as required by a band solver. Most times it requires a lot less core storage than other solvers. However, this involves the use of extra peripheral equipment not necessary in other type solvers (disc files or any type of supplementary storage).

Because of the compact nature of the front and because variables are eliminated as soon as possible, the operations on zero coefficients are minimised and the total arithmetic operations are fewer than with other methods.

On the other hand, an intricate housekeeping system is necessary for the frontal solution. However, since it is entirely performed with integer variables, little storage and computer time is used.

Lastly, the frontal solution method is definitely advantageous for solving large systems of equations when one is restricted by limits on core storage available.

3.11 Recent Developments of the Frontal Concept

A considerable amount of research on the frontal concept has been done over the past decade since its introduction by Irons [21] in 1970. This has led to the consolidation of the concept and a great variety of problems can be solved based on the frontal concept ideas [25]. Although some of these are not used in this work, an overview of the frontal concept recently developed capabilities is now presented.

The flexibilities inherent in the longevity-based frontal concept of Irons include diverse applications such as Guyan-reduced superelement generation [30], sub-structuring [25], matrix storage in frontal form [25], frontal matrix multiplication [25], frontal solution of unsymmetric matrices [29], planned disposition of variables within the front on the basis of natural longevity [27], dynamic allotment of destinations to solve the problem of non-compactness of the front [28], Eigen solution subspace iteration [25], and Eigen solution by determinant search and Sturm sequence check[26]. More about this last application will be said in Chapter 5 of this thesis.

FINITE ELEMENT FORMULATION IN DYNAMIC ANALYSIS

4.1 Basic Equations for Free Dynamic Vibration Problems

In the equilibrium problem presented earlier, conditions were assumed not to vary with time. However, there is a large range of practical problems in which the time dimension has to be considered. Examples are transient heat conduction, wave transmission in fluids and most important to structural analysts, the dynamic behaviour of structures. All these problems are identically formulated and the difference arises in the mathematical structure of the governing equations resulting from a variety of physical situations.

It has been shown [1] that as a result of semi-discretization many time-dependent problems can be reduced to a system of ordinary matrix differential equations of the form

$$M\ddot{\delta} + C\dot{\delta} + K\delta + F = 0 \quad (4.1)$$

where, in dynamic analysis, the four terms on the left hand side correspond to inertia, damping, static and forcing effects, respectively,

δ , $\dot{\delta}$, $\ddot{\delta}$ are the displacements, their first and their second derivatives,

M is a mass matrix,

C is a damping matrix

K is the stiffness matrix (already defined in static analysis),
and F is a forcing matrix.

M , C and K are symmetric and positive definite. This statement is accepted here and proved in references [1, 11, 12].

One of the particular cases of dynamic analysis arises where no damping or forcing terms exist and the Free Dynamic Vibration Problem expressed by

$$M\ddot{\delta} + K\delta = 0 \quad (4.2)$$

is then established.

If the general solution of this equation is

$$\delta = \bar{\delta} e^{i\omega t} \quad (4.3)$$

then on substitution one finds that ω can be calculated from

$$(-\omega^2 M + K)\bar{\delta} = 0 \quad (4.4)$$

This is an eigenvalue problem and for non-trivial solutions the following determinant must be equal to zero

$$\det(M - \lambda K) = 0 \quad (4.5)$$

where $\lambda = 1/\omega^2$.

Equation (4.5) will yield n values of λ (or λ_j , $j = 1, n$) if the size of the matrices M and K is $n \times n$. Since K and M are positive definite, all the roots of equation (4.5) are real positive numbers [36]. These roots are called the natural frequencies, ω_j , of the structural system. However, the solution of (4.5) does not yield the actual values of the displacements δ , but one can find n vectors $\bar{\delta}_j$ (corresponding to ω_j) which give the proportions of the various terms of δ . These vectors are called the natural nodes of vibration of the structural system.

There are several important properties of the eigenvalue problem which are used in its solution and adequately covered in references [36 - 38].

The identification of the free dynamic vibration problem has been con-

sidered; let us now consider how the new quantity involved, namely mass, can be formulated.

The virtual work equation, including its inertia and static terms, can be written as [14]

$$\int_V d\delta^T \rho H \ddot{\delta} dV + \int_V d\epsilon^T \sigma dV = 0 \quad (4.6)$$

$$\text{where } \delta = \sum N_i \delta_i \quad (4.7)$$

$d\delta$ is the virtual displacement

$d\epsilon$ is the virtual strain vector

and σ is the stress resultant vector.

Considering the work done by inertia forces

$$dW_{\text{inertia}} = \int_V d\delta^T \rho H \ddot{\delta} dV \quad (4.8)$$

and substituting equation (4.7) into equation (4.8) one gets

$$dW_{\text{inertia}} = d\delta^T \int_V N^T \rho H N dV \ddot{\delta} = d\delta^T M \ddot{\delta} \quad (4.9)$$

and thus the mass matrix for a line element (where the integration is over the element length) is

$$M = \int_x N^T \rho H N dx \quad (4.10)$$

where ρ is a mass density matrix

and H is a cross-sectional property matrix.

4.2 Element Characteristics and Assumptions

The element characteristics and assumptions established in section 2.2 and regarding cross-sectional variations, parabolic isoparametric formulation, types of deformation considered, cross-sectional behaviour, etc., are still valid in the case of the element formulation in dynamic

analysis. However, a new aspect is included in the dynamic analysis formulation. This regards the possibility of including or excluding rotatory inertia effects which are dealt with in section 4.4.

4.3 Element Definition

Again the element definition established in sections 2.3, 2.4, 2.12 and 2.13 is adopted in the dynamic analysis formulation. This regards the degrees of freedom of the element, the coordinate systems associated with the element and their interrelationship, the shape functions associated with the element nodes, the non-prismatic feature and the geometrical positioning of the element internal node.

4.4 Cross-sectional Properties Matrix

Rewriting the mass matrix equation of (4.10) as the element mass matrix, the typical contribution associated with nodes i and j is expressed as

$$M_{ij}^e = \int N_i^T \rho H N_j dx \quad (4.11)$$

$$= \rho \int N_i^T(\xi) H(\xi) N_j(\xi) \det J(\xi) d\xi \quad (4.12)$$

since ρ , the mass density, is assumed to be constant within the element.

Due to the fact that the element is allowed to be non-prismatic, the cross-sectional properties matrix H is a function of ξ . It is given (in mass formulation) by [7, 9, 31, 32, 39]

$$H_R(\xi) = \begin{bmatrix} A(\xi) & & & & \\ & J(\xi) & & & \\ & & A(\xi) & & \\ & & & I_y(\xi) & \\ & 0 & & & A(\xi) \\ & & & & & I_z(\xi) \end{bmatrix} \quad (4.13)$$

if rotatory inertia effects are to be included; or given by

$$H_{NR}(\xi) = \begin{bmatrix} A(\xi) & & & & \\ & 0 & & & \\ & & A(\xi) & & \\ & & & 0 & \\ & 0 & & & A(\xi) \\ & & & & & 0 \end{bmatrix} \quad (4.14)$$

if rotatory inertia effects are to be excluded.

4.5 Mass Density Matrix

This is a simple, constant and diagonal matrix involving mass density and is material related.

$$\rho = \begin{bmatrix} \rho & & & & \\ & \rho & & & \\ & & \rho & & \\ & & & \rho & \\ & 0 & & & \rho \\ & & & & & \rho \end{bmatrix} \quad (4.15)$$

4.6 Shape Functions Matrix

From equation (4.12) it is obvious that this matrix is also a function of ξ and its i th component is expressed as

$$N_i(\xi) = \begin{bmatrix} N_i(\xi) & & & & \\ & N_i(\xi) & & & \\ & & N_i(\xi) & & \\ & & & N_i(\xi) & \\ & 0 & & & N_i(\xi) \\ & & & & & N_i(\xi) \end{bmatrix} \quad (4.16)$$

The complete shape functions matrix is given by

$$N = [N_1, N_2, N_3] \quad (4.17)$$

4.7 Schemes for Evaluating Element Mass Matrix

The general problem of mass lumping and related processes in the finite element method has been investigated by many researchers [1, 31, 33, 34, 40]. It is perhaps useful to quote Hinton, Rock and Zienkiewicz [33] on their thoughts regarding the problem of mass lumping. Referring to the solution of the equations of dynamic analysis they state: "In the solution of such equations it would be extremely useful from the computational viewpoint if the matrices M and C could be made diagonal. In the early days the engineer has (apologetically) lumped his masses or forces by physical reasoning alone and appeared glad when a proper discretization procedure introduced him to consistent mass/force approximations of the form given by equation (4.10)[1, 40]. In recent years there has been a partial return to mass lumping as investigators found that the use of consistent masses did not always lead to improved accuracy and always involved additional computational work. Clough [41], Washizu [42] and others [43] have demonstrated this point with the use of simple elements for which lumping procedures are physically obvious".

This is certainly true for the family of quadratic isoparametric elements as shown for plate elements by Hinton and Bicanic [31] and as shown for beam elements in Chapter 6 of this work.

In conclusion, one can say there are a number of different schemes for evaluating M_{ij}^e which basically amount to ways of calculating values of the integral [31]

$$I_{ij}^e = \int N_i(\xi)^T N_j(\xi) d\xi \quad (4.18)$$

appearing in

$$M_{ij}^e = \rho H(\xi) \det J(\xi) I_{ij}^e \quad (4.19)$$

Three main approaches are possible [31] as follows:

I Lumped

(i) Equal Lumping

(ii) Special Lumping based on consistent mass matrix with Full Integration.

(iii) Special Lumping based on consistent mass matrix with Reduced Integration.

II Consistent

(i) Full Integration.

(ii) Reduced Integration.

(iii) Substitute Shape Functions.

III Diagonalised-Consistent

(i) First Order Lobatto Rule.

(ii) Second Order Lobatto Rule.

These are now investigated for the simple case of a prismatic member.

4.7.1 Lumped Matrix

The crudest possibility is to use equal lumping between the three element nodes and thus

$$\text{diag } I_E^e = (1/3, 1/3, 1/3) \quad (4.20)$$

each node being allocated a third of the element mass and the total element mass being equal to 1. Note that in equation (4.20) and following equations, one is referring to only one of the degrees of freedom for simplicity reasons, the procedures for the other degrees of freedom being, obviously, identical.

The other possibilities are special lumping based on the consistent mass matrix. Here, it should be noted that since

$$N_i = 1 \text{ at } x_i \quad (4.21a)$$

$$N_i = 0 \text{ at } x_j \quad (4.21b)$$

$$\text{and } \sum N_i = 1 \text{ at any } x \quad (4.21c)$$

the procedure of lumping offers infinite possibilities and what is recommended [33] is to compute the diagonal terms of the consistent mass matrix and then scale these terms so as to preserve the total mass of the element. For the case of special lumping with full integration (3 point Gauss-Legendre) one gets

$$\text{diag } I_{SL3}^e = (1/6, 4/6, 1/6) \quad (4.22)$$

and with reduced integration (2 point Gauss-Legendre)

$$\text{diag } I_{SL2}^e = (1/6, 4/6, 1/6) \quad (4.23)$$

the result is the same.

4.7.2 Consistent Matrix

If a consistent mass scheme (equation (4.11)) is adopted with full integration, then the resulting matrix of coefficients is

$$I_{C3}^e = \begin{bmatrix} 4/30 & 2/30 & -1/30 \\ 2/30 & 16/30 & 2/30 \\ -1/30 & 2/30 & 4/30 \end{bmatrix} \quad (4.24)$$

On the other hand if reduced integration is used, then

$$I_{C2}^e = \begin{bmatrix} 2/18 & 2/18 & -1/18 \\ 2/18 & 8/18 & 2/18 \\ -1/18 & 2/18 & 2/18 \end{bmatrix} \quad (4.25)$$

Note that the coefficients in each case sum to unity.

The case of a consistent scheme based on substitute shape functions is given by Hinton et al [33] and is not discussed in this work.

4.7.3 Diagonalised Consistent Matrix

The Lobatto rule forms the basis of this approach. The first two one-dimensional (line element) Lobatto rules are nothing more than the trapezoidal rule and Simpson's rule respectively [9]. Thus for the

case of a first order Lobatto rule one obtains

$$\text{diag } I_{DC1}^e = (1/2, 0, 1/2) \quad (4.26)$$

while for the second order Lobatto rule

$$\text{diag } I_{DC2}^e = (1/6, 4/6, 1/6) \quad (4.27)$$

It should be made clear that the coefficient matrices explicitly developed in this section refer to a prismatic and undistorted element. However, when considering a non-prismatic (Matrix H becomes a variable) and/or distorted ($\det J$ becomes a variable) element, it is not possible to derive coefficient matrices explicitly.

4.8 Space Transformations and System Mass Matrix

The element mass matrix derived in section 4.7 is referred to the element cartesian axes. It has to be transformed into the global system coordinates as explained in section 2.8 of this work. However, one should realize that the mass density matrix ρ is a concept meaningful only at global level. Therefore, the following procedure has to be carried out in order to obtain the element mass matrix at global system level:

- (i) Evaluate the integral

$$I^e = \int N^T(\xi) H(\xi) N(\xi) \det J(\xi) d\xi \quad (4.28)$$

at local element level, then

- (ii) Transform the integral I^e to global system level as follows:

$$I^S = [T^e]^T I^e T^e \quad (4.29)$$

- (iii) and finally the global system mass matrix is given by

$$M^S = \rho I^S \quad (4.30)$$

The reasoning behind assigning a global level meaning to the mass density matrix is that, if a structure wants to displace in a certain direction, or rotate about a particular axis (displacements and rotations are interpreted in the global axes set) it has to carry a mass with it and thus the mass density matrix must have a meaning in the global system so as to be compatible with the definition of displacements and rotations.

4.9 A Basic Dynamic Condensation of Unwanted Global Degrees of Freedom

In order to reduce the number of degrees of freedom to be solved for in vibration analysis, bearing in mind that an in-core solution technique is used (Chapter 5), all the unwanted global degrees of freedom are eliminated from the stiffness and mass matrices simply by deleting the relevant rows and columns. As an example, if a plane problem is to be solved, then one is only interested in 2 translational and 1 rotational global degrees of freedom and the 3 remaining global degrees of freedom are eliminated by this 'dynamic condensation'. Computationally, the specification of the global degrees of freedom to be eliminated from the matrices is achieved through user definition of the mass density matrix, i.e. if a global degree of freedom is not wanted in the dynamic analysis, then the corresponding diagonal term in the mass density matrix is assigned a value of zero. It becomes now more clear why the mass density matrix has to be defined in the global system.

It must be stressed that this type of condensation bears no relationship to the classical static condensation [50] or Guyan Reduction [30], which is not incorporated in NONPRI at this stage of development.

CHAPTER 5

SOLUTION OF EQUATIONS OF DYNAMIC ANALYSIS - A STURM SEQUENCE APPROACH

5.1 Introduction

Section 4.1 introduced the eigenvalue problem and the statement of its solution. However, at that stage, it was not discussed how to obtain the required eigenvalues and eigenvectors. Thus, it is the aim of this Chapter to review the most important techniques for eigenproblem solution and to discuss the technique adopted in this work - a Sturm sequence approach.

An important class of eigenvalue problems is defined by the following matrix form

$$A\ddot{\delta} + B\dot{\delta} = 0 \quad (5.1)$$

which, when substituting $\ddot{\delta}$ for

$$\ddot{\delta} = -\omega^2 \delta \quad (5.2)$$

becomes the characteristic value problem

$$(B - \omega^2 A)\delta = 0 \quad (5.3)$$

where B is symmetric, banded and positive definite; A is symmetric, banded (or diagonal); and δ is the associated eigenvector. Non-trivial solutions of equation (5.3) exist for a set of eigenvalues λ to be determined from

$$\det(A - \lambda B) = 0 \quad (5.4)$$

$$\text{where } \lambda = 1/\omega^2. \quad (5.5)$$

These eigenvalue equations arise from a variety of engineering problems such as [44]

- vibration analysis (subject of this work) where

$$A = M \text{ (mass matrix)}$$

$$B = K \text{ (stiffness matrix)}$$

$$\text{and } \lambda = 1/\omega^2 \text{ } (\omega \text{ being the system natural frequency}) \quad (5.6)$$

- structural stability analysis where

$$A = K_G \text{ (geometrical stiffness matrix)}$$

$$B = K_E \text{ (elastic stiffness matrix)}$$

$$\text{and } \lambda = 1/\mu \text{ } (\mu \text{ being a compressive stress factor}) \quad (5.7)$$

- vibration analysis of stretched structures where

$$A = M$$

$$B = K_E - K_G$$

$$\text{and } \lambda = 1/\omega^2 \quad (5.8)$$

5.2 Review of Solution Methods For Eigenvalue Problems

Let us consider the vibration analysis problem of equation (4.4) or (5.6) which can be rewritten as

$$K\delta = \lambda M\delta \quad (5.9)$$

and specifically the determination of the lowest eigenvalues λ_j and associated eigenvectors δ_j .

The available methods of solution can be classified in four categories [4] according to the basic property employed as the basis for the solution algorithm. In the first category are the vector iteration methods in which the property used is

$$K\delta_{i+1} = \lambda_i M\delta_i \quad (5.10)$$

The second category consists of the transformation methods using

$$\delta^T K \delta = \Lambda \quad (5.11)$$

and

$$\delta^T M \delta = I \quad (5.12)$$

where

$\delta = (\delta_1, \delta_2, \dots, \delta_n)$ a matrix of eigenvectors

$\Lambda = \text{diag}(\lambda_i), i=1, 2, \dots, n$

and I = identity matrix (diagonal).

Polynomial iteration techniques form the third category which is based on

$$p(\lambda_i) = 0 \quad (5.13)$$

where

$$p(\lambda) = \det(K - \lambda M) \quad (5.14)$$

In the fourth category one finds methods based on the Sturm sequence property of the characteristic polynomials

$$p(\lambda) = \det(K - \lambda M) \quad (5.15)$$

and

$$p_r(\lambda_r) = \det(K_r - \lambda_r M_r) ; r = 1, 2, \dots, n-1 \quad (5.16)$$

where $p_r(\lambda_r)$ is the r th characteristic polynomial associated with equation (5.9).

Perhaps not forming a different category of solution techniques since the basic ideas involved are not different, but certainly providing a different approach to the handling of equation (5.9) one finds a frontal based solver for vibration analysis [26].

Very important to realize is the fact that all solution methods have to be of an iterative nature since solving equations (5.9) is equivalent to finding the roots of $p(\lambda)$ and explicit formulas are not available for the case when the order of p is greater than 4 (which is always the situation because the order of p is the order of M and K).

5.2.1 Vector Iteration Methods

The basic equation considered here is

$$K\delta = \lambda M\delta \quad (5.9)$$

and the idea is to satisfy it by operating directly on it. Assuming a vector x_1 for δ and some value for λ , say unity, one can calculate the right hand side of (5.9) as

$$R_1 = 1.M.x_1 \quad (5.17)$$

Because x_1 is an arbitrary vector, one generally cannot take that $Kx_1 = R_1$. If this was the case then, x_1 would be the eigenvector and the assumption of x_1 would have been a very fortunate one. However, in the general case of $Kx_1 \neq R_1$ one obtains an equilibrium equation similar to that of static analysis, i.e.

$$Kx_2 = R_1, \quad x_2 \neq x_1 \quad (5.18)$$

where x_2 is the displacement corresponding to load R_1 . An iterative cycle is now set up and by performing it repeatedly one obtains better and better approximations to the eigenvector. This is, in fact, the procedure used in inverse iteration, but all the other types of iteration methods are of a similar nature.

5.2.2 Transformation methods

As stated in section 5.2 the basic properties used in these methods are

$$\delta^T K \delta = \Lambda \quad (5.11)$$

and

$$\delta^T M \delta = I \quad (5.12)$$

Since the $n \times n$ matrix δ which diagonalises matrices K and M is unique, one tries to find it by iterating. The procedure is to transform K and M to diagonal form by successive pre and postmultiplication using matrices P_k^T and P_k where $k = 1, 2, \dots$.

Once the process is initialised by making

$$K_1 = K \quad (5.19)$$

$$\text{and } M_1 = M \quad (5.20)$$

one continues by forming

$$\begin{aligned}
 K_2 &= P_1^T K_1 P_1 \\
 K_3 &= P_2^T K_2 P_2 \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 K_{k+1} &= P_k^T K_k P_k
 \end{aligned} \tag{5.21}$$

and similarly

$$\begin{aligned}
 M_2 &= P_1^T M_1 P_1 \\
 M_3 &= P_2^T M_2 P_2 \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 M_{k+1} &= P_k^T M_k P_k
 \end{aligned} \tag{5.22}$$

where P_k are chosen to make K_k and M_k closer to diagonal form. In the end, one requires that

$$\begin{aligned}
 K_{k+1} &\rightarrow \Lambda \\
 \text{and } M_{k+1} &\rightarrow I \text{ as } k \rightarrow \infty
 \end{aligned} \tag{5.23}$$

and if ℓ is the last iteration, then

$$\delta = P_1 P_2 \cdots P_\ell \tag{5.24}$$

In practice, it is not required that M_{k+1} and K_{k+1} converge to I and Λ respectively, but only that they converge to diagonal form. Namely, if

$$\begin{aligned}
 K_{k+1} &\rightarrow \text{diag}(K_r) \\
 \text{and } M_{k+1} &\rightarrow \text{diag}(M_r) \text{ as } k \rightarrow \infty
 \end{aligned} \tag{5.25}$$

then

$$\Lambda = \text{diag} \left(\frac{K_r^{(\ell+1)}}{M_r^{(\ell+1)}} \right) \quad (5.26)$$

and

$$\delta = P_1 P_2 \dots P_\ell \text{diag} \left(\frac{1}{\sqrt{M_r^{(\ell+1)}}} \right) \quad (5.27)$$

The above ideas have originated methods such as Jacobi and Householder-QR methods [4,47].

5.2.3 Polynomial iteration techniques

As previously discussed the roots of the characteristic polynomial

$$p(\lambda) = \det (K - \lambda M) \quad (5.14)$$

are the eigenvalues of the eigenproblem $K\delta = \lambda M\delta$. One can therefore operate on $p(\lambda)$ to find the roots and thus obtain the eigenvalues. For this several procedures have been proposed [47-49], but actually there are two basic approaches: explicit and implicit evaluation procedures, both of which may use the same basic iteration schemes. It is assumed that the solution is performed directly using K and M , i.e. no transformation of the problem to a different form is carried out. This is so because, if only a few eigenvalues are required for calculation, operating directly on K and M turns out to be most effective [4].

When using a polynomial iteration method, only the eigenvalues are calculated and the associated eigenvectors are then obtained by inverse iteration with shifting.

For explicit polynomial iteration methods, one starts by writing $p(\lambda)$ as

$$p(\lambda) = a_0 + a_1 \lambda + a_2 \lambda^2 + \dots + a_n \lambda^n \quad (5.28)$$

and evaluating the coefficients $a_0, a_1, a_2, \dots, a_n$. Then one calculates the roots of the polynomial. However, there is an important shortcoming in this method in that small errors in the coefficients $a_0, a_1, a_2, \dots, a_n$ originate large errors in the roots of the poly-

nomial. Since these small errors are almost unavoidable (because of computer round-off error) this method has almost been abandoned [4].

For an implicit polynomial iteration solution one evaluates $p(\lambda)$ directly, i.e. not calculating the coefficients a_0, a_1, \dots, a_n first. The value of $p(\lambda)$ is obtained by decomposing $K - \lambda M$ into a lower unit triangular matrix L and an upper triangular matrix S , i.e.

$$K - \lambda M = LS \quad (5.29)$$

and subsequently

$$\det(K - \lambda M) = \prod_{i=1}^n S_{ii} \quad (5.30)$$

5.2.4 Methods Based on the Sturm Sequence Property

The Sturm sequence property [4] of the characteristic polynomials of the problem $K\delta = \lambda M\delta$ and its associated constraint problems is the basis of these methods. The important result used here is as follows: assuming that for a shift μ_k , the Gauss factorization of $K - \mu_k M$ into LDL^T (D is a diagonal matrix and L is a lower triangular matrix) can be achieved, then the number of negative elements in D is equal to the number of eigenvalues smaller than the shift μ_k . This is used to develop a solution algorithm [4, 44, 45]. As for polynomial iteration techniques, the solution is carried out by operating directly on K and M . Also, in this case, only the eigenvalues are solved for and the eigenvectors have to be found by inverse iteration with shifting.

Consider that one requires the solution for all eigenvalues between λ_ℓ and λ_u , where λ_ℓ and λ_u are the lower and upper limits, of the interval respectively. This situation is shown on figure 5.1. The solution procedure is based on the triangular factorization of $K - \mu_k M$ where μ_k is chosen so as to obtain from the positive or negative signs of the diagonal elements in the factorized form meaningful information about the required eigenvalues.

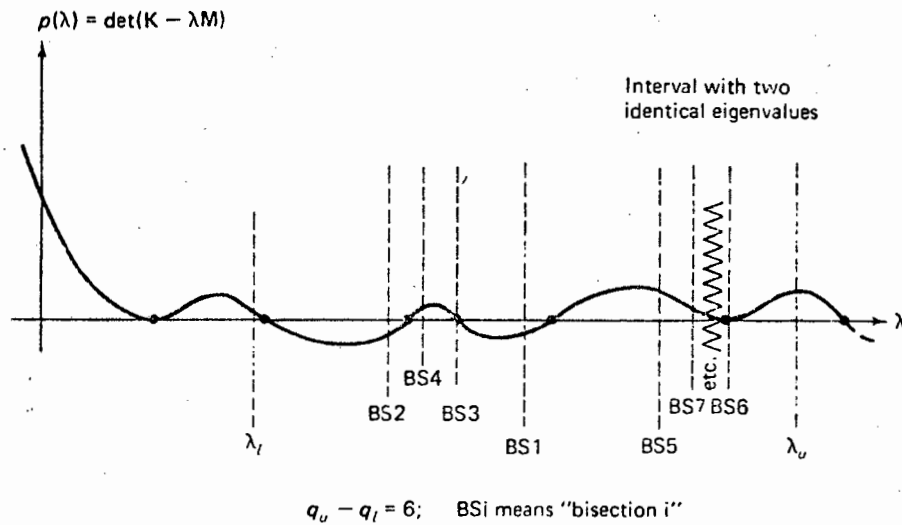


Figure 5.1 Use of Sturm sequence property and bisection to locate eigenvalues.

The Bisection method procedure is now described:

- (i) Factorize $K - \lambda_\ell M$ and find how many eigenvalues, say q_ℓ , are smaller than λ_ℓ
- (ii) Similarly, for λ_u , find q_u . Therefore, there are $q_u - q_\ell$ eigenvalues between λ_ℓ and λ_u .
- (iii) Using a simple bisection procedure identify intervals in which the individual eigenvalues lie (refer to figure 5.1). In this process, those intervals in which more than one eigenvalue are known to exist are successively bisected and the Sturm sequence check is performed until all eigenvalues have been isolated.
- (iv) Calculate the eigenvalues to a required accuracy and finally,
- (v) Obtain the associated eigenvectors by inverse iteration.

Note that in (iv) one usually switches from bisections to a more efficient search procedure.

5.2.5 A Frontal Based Method with Static Condensation

Recent work on extending the capabilities of the frontal solution technique [21] to perform eigenvalue/eigenvector solutions has been carried out by Cedolin and Gallagher [26]. It is especially interesting since a Guyan [30] or static condensation [50] scheme has been incorporated in the algorithm. Broadly, the procedure employed makes use of different aspects of the frontal solver to

- (i) carry out a condensation of the global degrees of freedom with which there is no mass associated;
- (ii) perform a determinant search of the reduced system of equations in order to calculate the eigenvalues, and
- (iii) perform a final calculation of the complete eigenvectors.

Let us now discuss each of the operations involved. The eigenproblem of equation (5.9)

$$K\delta = \lambda M\delta \quad (5.9)$$

is considered, the terms involved having previously been defined.

Assuming that the mass of the system may be lumped at 'master' degrees of freedom, δ_a , with a diagonal mass matrix M_a , equations (5.9) can be partitioned as

$$\begin{bmatrix} K_{cc} & K_{ca} \\ K_{ac} & K_{aa} \end{bmatrix} \begin{Bmatrix} \delta_c \\ \delta_a \end{Bmatrix} = \begin{Bmatrix} 0 \\ \lambda M_a \delta_a \end{Bmatrix} \quad (5.31)$$

where the subscript c refers to condensed degrees of freedom and subscript a refers to active degrees of freedom. Now, instead of the usual procedure of expressing $\{\delta_c\}$ in terms of $\{\delta_a\}$ through the upper partition of equation (5.31), obtaining the reduced eigenproblem

$$[K_a]\{\delta_a\} = \lambda[M_a]\{\delta_a\} \quad (5.32)$$

$$\text{with } [K_a] = [K_{aa}] - [K_{ac}]^{-1}[K_{cc}][K_{ca}] \quad (5.33)$$

the frontal solution offers a natural way of partitioning the system of equations (5.9) in the form of equations (5.31). It then deals with the active degrees of freedom δ_a without even computing the reduced matrix $[K_a]$. This is achieved as follows: the pre-frontal analysis of the system topology dictates the elimination of a specific degree of freedom as soon as the last element containing it is assembled (refer to Chapter 3). If then 'dummy' elements having no stiffness and nodal loads equivalent to $\lambda[M_a]\{\delta_a\}$ are added to the system with a higher sequence number, having as degrees of freedom the active ones, these will be the last to be eliminated. Thus, the first part of the frontal procedure (assemblage of the regular elements) is equivalent to the elimination of the condensed degrees of freedom δ_c and since these are independent of the unknown λ , needs not to be repeated in the iterative procedure. For this, the working space in the algorithm is memorised at the stage at which the 'dummy' elements start to be assembled, and it is reproduced at anytime the calculation is reinitiated with a new value of λ .

The next step is the determinant search for eigenvalues. In the case of vibration analysis the 'nodal loads' $\lambda[M_a]\{\delta_a\}$ are unknown since they depend on the eigenvalues λ_i and eigenvectors δ_{ai} . However, equation (5.31) is rewritten

$$\begin{bmatrix} K_{cc} & K_{ca} \\ K_{ac} & K_{aa} - \lambda M_a \end{bmatrix} \begin{Bmatrix} \delta_c \\ \delta_a \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \quad (5.34)$$

and this system can be handled with the frontal algorithm by defining λM_a as the stiffness of the 'dummy' elements. The problem is now the one of finding the roots of the determinant of the matrix on the left hand side of (5.34), this being once again solved by an iterative scheme. The value of the determinant for a particular λ value is calculated as the product of the pivots, once a Gaussian elimination has been performed. The repetition of this process is limited to the 'dummy' elements. A back substitution according to equation (5.34) is carried out after each elimination cycle and the result may be taken as an approximation to the wanted eigenvector.

Finally, when the approximations λ_i, δ_{ai} to all the required eigenpairs have been found (with a specified accuracy), they are used to construct the right hand sides of equation (5.31) and a final backsubstitution (now not limited to the dummy elements) provides the final and complete eigenvectors associated with each eigenvalue. The complete procedure can be best visualised by examining figure 5.2 [26].

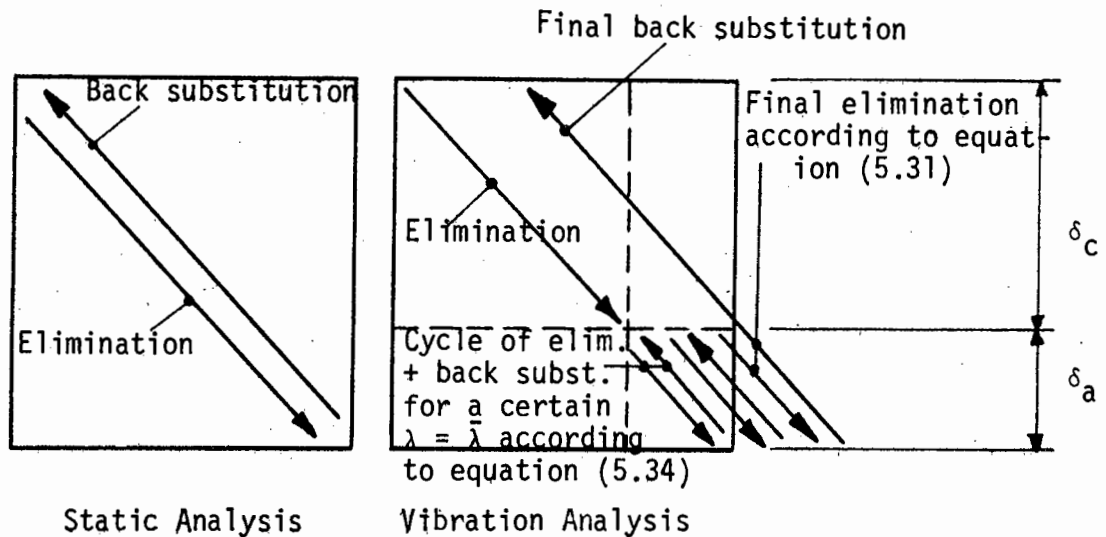


Figure 5.2 Representation of static and vibration analyses as performed by the frontal approach. (From Cedolin and Gallagher [26]).

5.3 Choice Of Solution Method

In section 5.2 the basic solution techniques for eigenproblems were outlined. However, one must realize that no approach is always and unconditionally the most efficient, and one would rather think in terms of selecting a particular technique for each specific problem. This is not practically possible and therefore instead of choosing one of the methods for general use, one should try to couple the basic

solution methods in order to design effective techniques which are so crucial in the final evaluation of the competitiveness of a structural analysis program.

Bearing the above considerations in mind there were, to my knowledge, four techniques worth investigating for possible use in NONPRI, as follows:

- (i) Determinant Search Method [4, 53]
- (ii) Subspace Iteration Method [4, 53]
- (iii) Frontal Based Method [26] and
- (iv) Sturm Sequence Based Method [44, 45, 46].

Not feeling qualified enough to influence the reader as to which method is the 'best', I would rather prefer to abstain from my opinion and direct the reader to the very useful thoughts on this topic expressed by Bathe and Wilson [4].

Nevertheless, the Sturm sequence/Bisection/Inverse Iteration algorithm presented by Gupta [44] is adopted in this work for its effectiveness, clarity, ease of implementation and degree of control allowed to the user when dealing with practical applications. It is hoped this will become clear in the sections to follow.

5.4 The Adopted Sturm Sequence/Bisection/Inverse Iteration Algorithm

Many structural systems are characterized by highly banded mass and stiffness matrices. Also, in the case of lumped mass models, the mass matrix becomes diagonal. Therefore, it is obvious that exploiting the banded forms of these matrices will be advantageous. However, the standard eigenproblem solution routines do not take advantage of this special matrix form and are therefore inefficient. These standard routines require the reduction of equation (5.3) to the transformed form

$$(C - \lambda I)\delta = 0 \quad (5.35)$$

where I is an identity matrix

and C is generally a full matrix form.

The inefficiency of the methods used for the solution of (5.35) arises primarily from the full nature of C .

The adopted Sturm sequence based method calculates the roots directly from the banded M and K matrices, and the associated modes of vibration are then calculated by inverse iteration. Furthermore, the adopted method allows one to calculate either all the roots or any specific roots without having to calculate any other. The Sturm algorithm used in NONPRI will calculate all the roots or the first required x number of largest roots λ (i.e. the x number of smallest natural frequencies ω) lying in a specified interval λ_l, λ_u .

5.4.1 Sturm Sequence Property

The leading principal minors $f_r(\lambda)$ of $M - \lambda K$ have the Sturm sequence property [45, 52] as previously mentioned in this work. Therefore one can say that the number of sign changes of consecutive elements of the sequence $f_r(\lambda)$, starting with $f_0(\lambda) = 1$ is equal to the number or eigenvalues of $M - \lambda K$ smaller than λ in value. Fortunately, since only signs of $f_r(\lambda)$ are needed, this can be obtained without computing the actual values of $f_r(\lambda)$. It is an eigenproblem solution routine based on this property [44] that has been adopted in the present work. The procedure consists of a main program STURM and four subroutines EIGNUM, BISECT, LOCATE and EIGVEC.

5.4.2 The Bisection Procedure

Once the range of values for λ , i.e. (λ_l, λ_u) , has been specified, this procedure uses EIGNUM to calculate the number of roots lying in (λ_l, λ_u) , and then the wanted roots are isolated to a certain specified accuracy by successive bisections performed by BISECT. The bisection technique used in this program simultaneously finds the upper and lower limits of all required roots at a particular stage. Consequently, successively

smaller bounds are obtained during each stage of the procedure, thereby effecting fast convergence.

Once the individual roots are isolated, they are located accurately by using LOCATE, which performs successive linear interpolations between weighted function values of opposite signs. It is now necessary to actually compute the values of the determinants $f_r(\lambda)$, which can have very large (causing machine overflow) or very small (causing machine underflow) values and thus the determinants are stored as double precision quantities in the machine. When LOCATE is used, convergence becomes slow if the determinant values corresponding to upper and lower limits of the isolated roots, obtained by BISECT, have a very large difference in value. When this occurs it is better to reduce such limiting bounds for each root by further bisection, the result being faster convergence.

5.4.3 The Computational Procedure

The present algorithm requires a relatively small working storage D of size $(m+1)(2m+1)$ where m is the semibandwidth of the stiffness matrix K. Additionally it requires the storing of the upper symmetric parts of K and M, which is done in rectangular arrays with the diagonal elements along the first column [24].

- Routine EIGNUM

The important part of the algorithm regarding the determination of the leading principal minors (subroutine EIGNUM) employs a special variant of Gaussian elimination with pivoting to preserve the information necessary to evaluate signs of the required determinants and to ensure the multipliers do not exceed unity in modulus - therefore numerical stability is guaranteed [37]. The leading principal minors are obtained during the reduction procedure in which the $(r+1)$ th row is not involved until the rth major step. All the computations are done in working array D, which can be visualised as moving down the main diagonal at the end of the reduction of its $(m+1)$ th row - refer to figure 5.3.

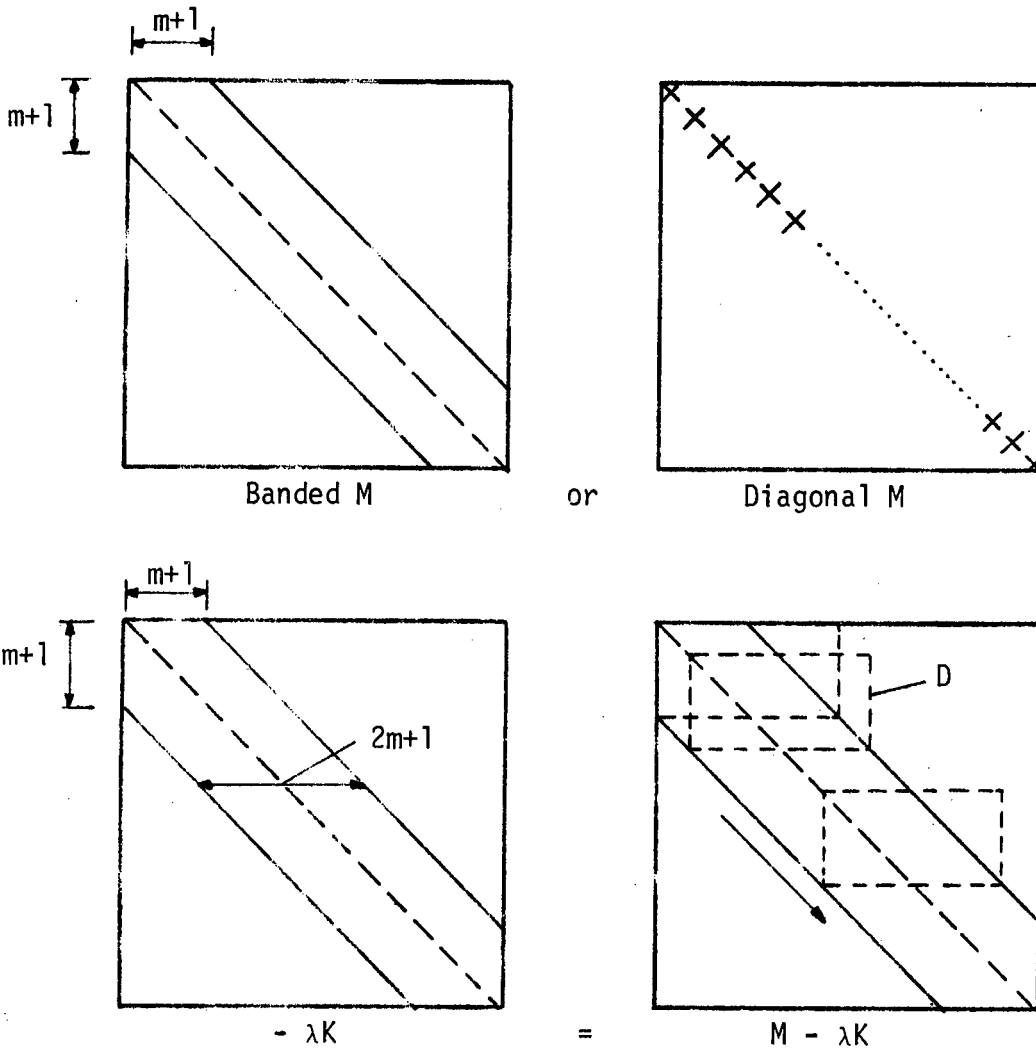


Figure 5.3 Arrangement of mass, stiffness and working matrices in numerical reduction scheme.

Initially, the first $m+1$ rows of $M - \lambda K$ are put into D and m major operations are performed in order to reduce the subdiagonal elements of D to zero. Denoting the elements of D by a_{ij} , the following operations are carried out for each value of i from 1 to r during a typical r th step:

1. Interchange $a_{r+1,j}$ and a_{ij} ($j = i, i+1, \dots, 2m+1$) if $|a_{r+1,i}| > |a_{ii}|$.

2. Replace $a_{r+1,j}$ by $a_{r+1,j} - (a_{r+1,i}/a_{ii})a_{ij}$
 $(j = i, i+1, \dots, 2m+1).$

The leading principal minor is given by

$$f_{r+1}(\lambda) = (-1)^N a_{11} a_{22} \dots a_{r+1, r+1} \quad (5.36)$$

in which N is the total number of interchanges occurring so far and a_{ii} is the current diagonal element. Then, for every value of r increasing from $m+1$ to $n-1$, n being the order of M and K , the rows of D are first rearranged as follows.

3. Replace the first row and move each of the next m rows up one row and to the left by one column.
4. Copy $(r+1)$ th row from M and K into the $(m+1)$ th row of D ,

$$D(m+1, j) = M(r+1, j) - \lambda K(r+1, j) \quad (j = 1, 2, \dots, 2m+1)$$

which is next followed by typical operations as below for each value of i from 1 to m .

5. Interchange $a_{m+1, j}$ and a_{ij} ($j = i, i+1, \dots, 2m+1$) if $|a_{m+1,i}| > |a_{ii}|$.
6. Replace $a_{m+1,j}$ by $a_{m+1,j} - (a_{m+1,i}/a_{ii})a_{ij}$
 $(j = i, i+1, \dots, 2m+1).$

The leading principal minor may then be computed as

$$f_{r+1}(\lambda) = (-1)^{F1} a_{11} a_{22} \dots a_{m+1, m+1} \quad (5.37)$$

where $F1$ stores the value $a'_{11} a'_{22} \dots a'_{r-m}$,

a'_{ii} being the previous first row diagonal elements of D , and a_{ii} being its current diagonal elements.

As previously stated, only the sign of $f_{r+1}(\lambda)$ is needed for the Sturm sequence, and this can be obtained without computing the determinant values of equations (5.36) and (5.37). Therefore, if the sign of $f_r(\lambda)$ is known, then each time step 1 or 5, as the case may be, is performed, this sign is switched only if both an interchange occurs and a_{ij} and $a_{r+1,i}$ have the same sign. Lastly, at the end of the r th major step, the sign is changed again if $a_{r+1, r+1}$ is negative.

- Routine BISECT

Subroutine BISECT isolates the individual roots. Thus, at any stage of its operation and for given upper and lower bounds, it computes

$$H = \frac{(\lambda_u^r + \lambda_l^r)}{2} \quad (5.38)$$

Assuming the existence of p eigenvalues in (λ_u^r, H) indicated by EIGNUM the routine continues as follows:

- (a) If $p = 0$, then repeat the procedure with $\lambda_u^r = H$.
- (b) If $p > 0$ and $|\lambda_u^r - H| < \text{EPS}$, then p repeated roots occur, each being equal to H in numerical value. Note that EPS is the specified precision of root separation during isolation of individual roots.
- (c) If $p > 0$ and $|\lambda_u^r - H| \geq \text{EPS}$, then H is the lower bound for λ_i , $\lambda_{i+1}, \dots, \lambda_{i+p-1}$ and the upper bound for $\lambda_{i-1}, \lambda_{i-2}, \dots$, provided their present upper bounds are already greater than H .

The above process is repeated and the bounds of all isolated roots are thus obtained.

- Routine LOCATE

Isolated roots are then accurately located by using subroutine LOCATE. At any particular stage of interpolation, if λ_{r+1} and λ_r are the given bounds such that $f(\lambda_r) = \det(M - \lambda_r K)$ and $f(\lambda_{r+1}) = \det(M - \lambda_{r+1} K)$ have

different signs, linear interpolation is performed between the weighted functions as

$$\lambda_{r+2} = \frac{[\lambda_{r+1} \cdot f(\lambda_r) - \lambda_r \cdot W_{r+1} \cdot f(\lambda_{r+1})]}{[f(\lambda_r) - W_{r+1} \cdot f(\lambda_{r+1})]} \quad (5.39)$$

where $W_{r+1} = 2^s$ and $s = \frac{1}{2}(p-1)(p-2)$,

where p is the number of times that λ_r has been used unaltered in the interpolation process. Then, the new value of λ_{r+1} is taken to be the old value of λ_r or λ_{r+1} , such that $f(\lambda_{r+2})$ and $f(\lambda_{r+1})$ have different signs. The process of weighted interpolation is successively repeated until the specified accuracy has been achieved.

5.4.4 The Inverse Iteration Procedure

- Routine EIGVEC

Once the roots have been accurately calculated, the associated eigenvectors are simply calculated by the inverse iteration procedure

$$(M - \lambda K) \delta^{(r+1)} = K \delta^{(r)} \quad (5.40)$$

where λ is the calculated eigenvalue and $\delta^{(r+1)}$ converges to the required eigenvector δ . This procedure starts with the assumption that $K \delta^{(1)}$ is a unit vector, after the subdiagonal elements of $M - \lambda K$ have been reduced to zero and it has been observed [44] that, at most, two iterations are necessary for the accurate calculation of δ . Equation (5.40) is solved by a special variant of Gaussian elimination with interchanges as described previously, followed by back substitution. When coincident eigenvalues are encountered, special techniques [37, 44] have to be used in order to calculate the associated eigenvectors.

Lastly, it should be noted that the inverse iteration procedure necessitates additional storage equal to $n(2m+1)$, since the procedure uses partial pivoting which results in row interchanges. The inverse iteration procedure is carried out in subroutine EIGVEC

5.5 Scope and Advantages of Adopted Algorithm

The common transformation methods, e.g. QR, Givens Householder, Jacobi, etc., need the characteristic equation to be reduced to the form of equation (5.35) in which C is a full matrix. This creates many limitations as far as storage required is concerned. On the other hand, the present procedure operating directly on the banded matrices M and K is capable of solving larger problems. The storage requirements are small since only a working array D of size $(m+1)(2m+1)$ is required. This is obviously additional to the normal storage required for the upper symmetric bands of M and K . The resulting procedure is fast and accurate. The computation time for the adopted procedure [44] is proportional to $2nm^2 \times NR \times F$ compared to n^3 for the common case, where n is the order of the system, m is the semibandwidth of K , NR is number of required roots and F is the average number of reductions performed by EIGNUM per root - normally between 10 and 15 [44].

Of course, the procedure incorporated in the program NONPRI has ultimately a limit since it is an in-core solution procedure. Therefore, when the need arises, it will have to be made an out-of-core algorithm at which stage one has to start paying the penalties associated with out-of-core procedures, namely considerably slower running times and a considerable increase in cost of runs.

CHAPTER 6

NUMERICAL EXAMPLES AND RESULTS

6.1 Introduction

Practical examples are now presented to show the advantages, flexibility of modelling and accuracy of the present formulation, as implemented in the program NONPRI. Results from other sources are given for comparison purposes. Running times are also given in some examples to demonstrate the kind of execution times to be expected from NONPRI.

The examples are arranged in two sections, the first dealing with static analyses and the second dealing with dynamic vibration analyses.

6.2 Static Analyses

6.2.1 Simply Supported Deep Beam

A deep beam with a span to depth ratio equal to 3, as shown in figure 6.1, is used to illustrate the deep beam capabilities of the present formulation. The mid span transverse displacement is chosen as a means of comparing the results obtained from the present element with results obtained from different analyses. These are as follows, and referred to in Table 6.1:

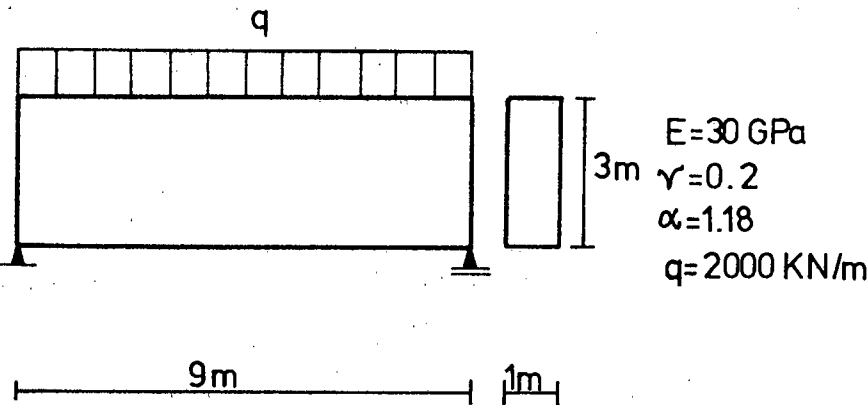


Figure 6.1 Simply supported Deep Beam.

- Analysis I : 2-noded shallow beam finite element
 Analysis II : 2-noded deep beam finite element
 Analysis III: 3-noded (NONPRI) deep beam finite element
 Analysis IV : Force method (beam idealisation)
 Analysis V : Parabolic isoparametric plane stress finite element.

Analysis	No. of Elements	Mid span transverse displacement (mm)	% Error w.r.t. IV	% Error w.r.t. V
I	6	2.5312	20.1	30.1
II	6	3.0607	3.4	15.5
III	2	3.1622	0.2	12.7
IV	-	3.1684	0.0	12.6
V	6	3.6235	-	0.0

Table 6.1

The above results clearly show not only the obvious importance of including shear deformation effects in deep beam analysis but also the fact that a considerably more accurate displacement is obtained with NONPRI using 2 elements than with the 2-noded deep beam element using 6 elements. This is due to the better approximation (quadratic shape functions) of the effective shear rotation ϕ used in the present formulation. In the 2-noded element formulation, the effective shear rotation is taken as constant within an element [54].

As expected, the beam idealisation solutions for the deep beam problem are fairly inaccurate when compared with the solution obtained when performing a 2-dimensional plane stress analysis. The values obtained for the bending moments and shear forces using the NONPRI element are exact.

6.2.2 Tapered Cantilever

The tapered cantilever with deep beam characteristics shown in figure 6.2 is chosen to demonstrate the accuracy of a non-prismatic analysis performed by NONPRI.

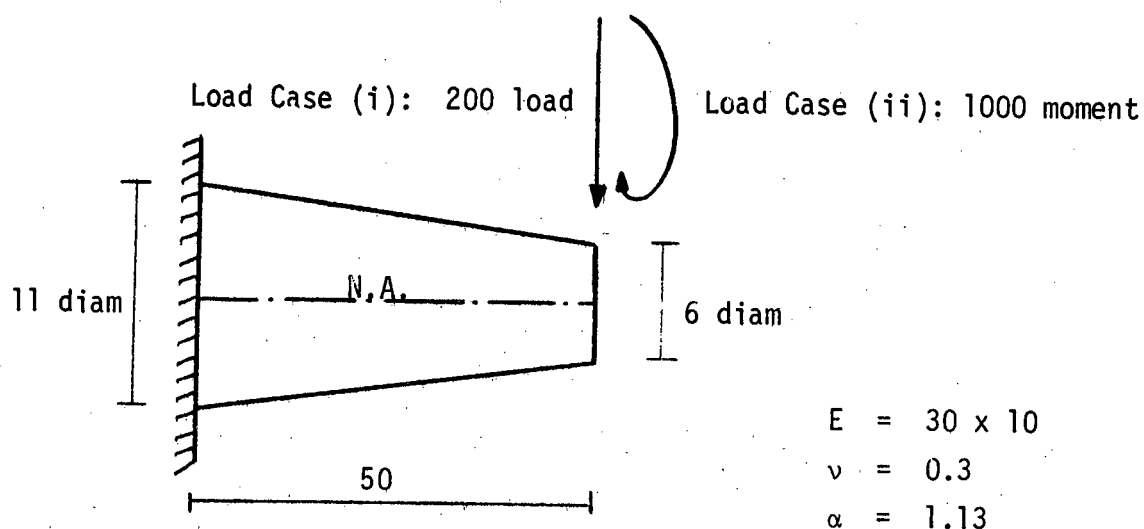


Figure 6.2 Tapered Cantilever with Circular cross-section

An exact solution and solutions using 2-noded prismatic elements are used for comparison. The variable cross-section is approximated by a stepped model in the case of the 2-noded prismatic elements, with the method of taper modelling being an area averaging one.

Two separate load cases are considered, namely

- (i) A point load of 200 units acting vertically down at the tip of the cantilever.
- (ii) A torsional moment of 1000 units also acting at the tip of the cantilever.

The results obtained are plotted in figure 6.3.

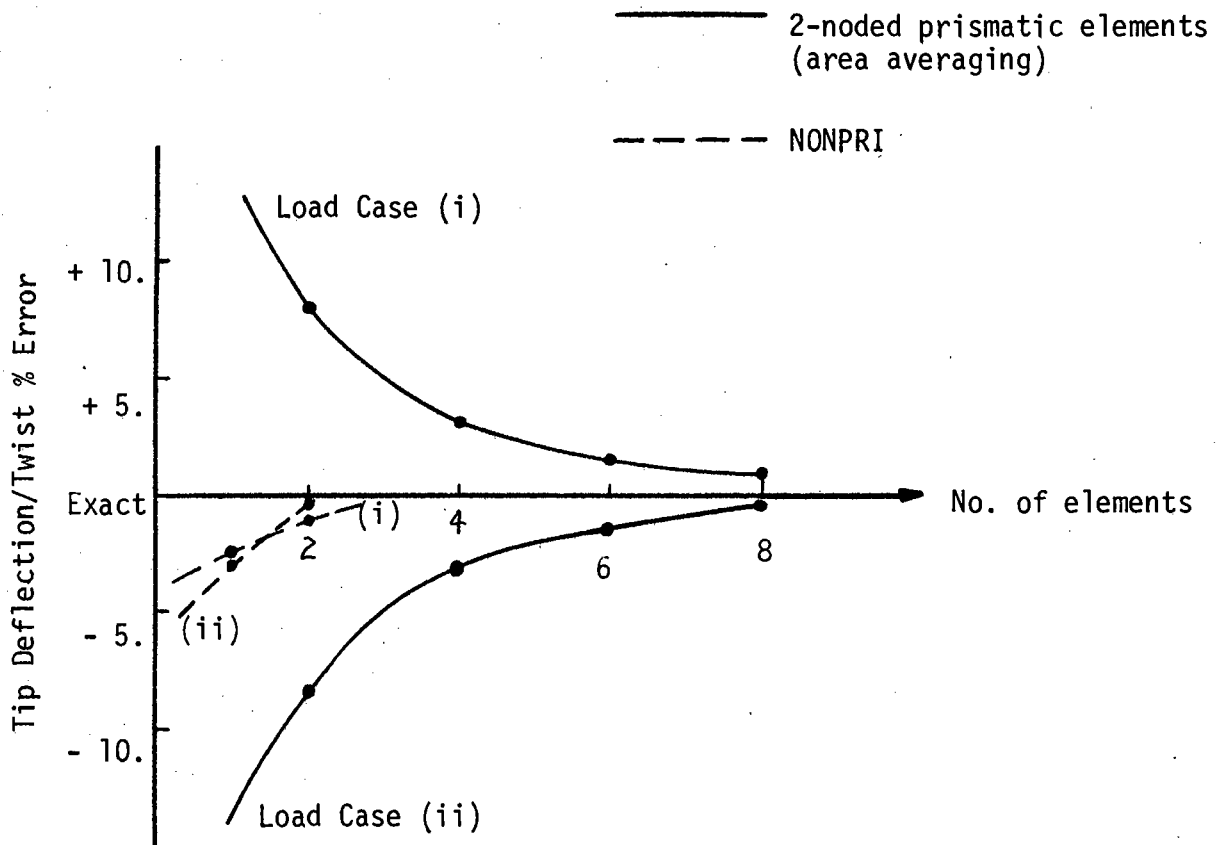
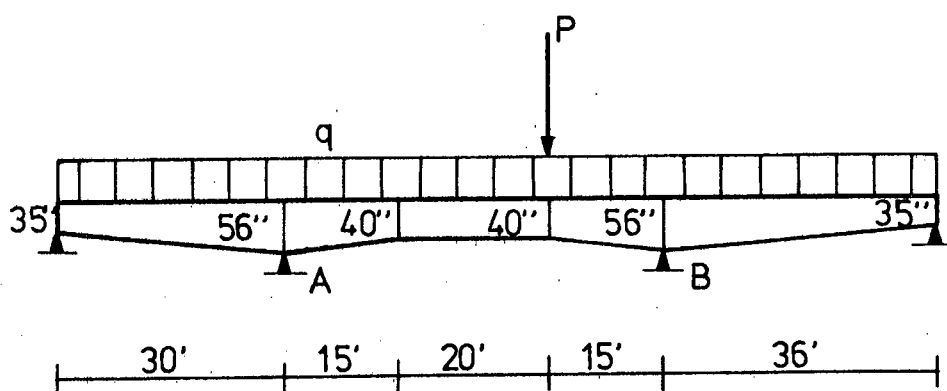


Figure 6.3 Tip deflection % error vs grid refinement

The advantages of the NONPRI non-prismatic formulation become obvious from figure 6.3. Bending moments, shear forces and torsional moments obtained by NONPRI are all exact.

6.2.3 Three-Span Non-Prismatic Beam

The third problem to be considered is one presented by Lightfoot [55] as shown in figure 6.4.



BEAM IS 24" WIDE , $P=20$ tons , $q=1$ ton/ft

Figure 6.4 Three-span non-prismatic continuous beam

This non-prismatic continuous beam was analysed 6 different times as follows:

Analysis I : 5 NONPRI 3-noded elements were used, 3 for the centre span and 1 each for the end span - this provides an exact modelling of the cross-section variation.

Analysis II : 3 NONPRI 3-noded elements were used, 1 for each span - this approximates the cross-section depth of the centre span by a parabola.

Note that in this analysis the centre span element has an off-centre internal node, in fact, it is positioned under the point load.

Analysis III: 18 2-noded elements were used, 6 in each span - this gives a steplike modelling of the cross-section profile in the tapering part of the structure.

Analysis IV : 24 2-noded elements were used, 8 in each span - the same as analysis III. It should be noted that in analyses III and IV an area averaging method of taper modelling is used.

Analysis V : Is a force method solution.

Analysis VI : The one presented by Lightfoot [55] - here he modifies the stiffnesses to account for taper in the spans.

Some relevant results are compared in Table 6.3.

Analysis	No. of elements	Support moments (ton-in)		Support reactions (ton)		Moment at point load (ton-in)	Normalised deflection under point load
		A	B	A	B		
I	5	2782.	3461.	53.92	67.97	1468.	1.002
II	3	2761.	3601.	54.43	65.26	1300.	1.163
III	18	2762.	3437.	55.15	65.73	1485.	1.054
IV	24	2779.	3464.	54.50	66.39	1470.	1.004
V	-	2782.	3468.	53.94	67.78	1463.	1.000
VI	-	2784.	3479.	54.10	67.26	1455.	0.995

Table 6.3

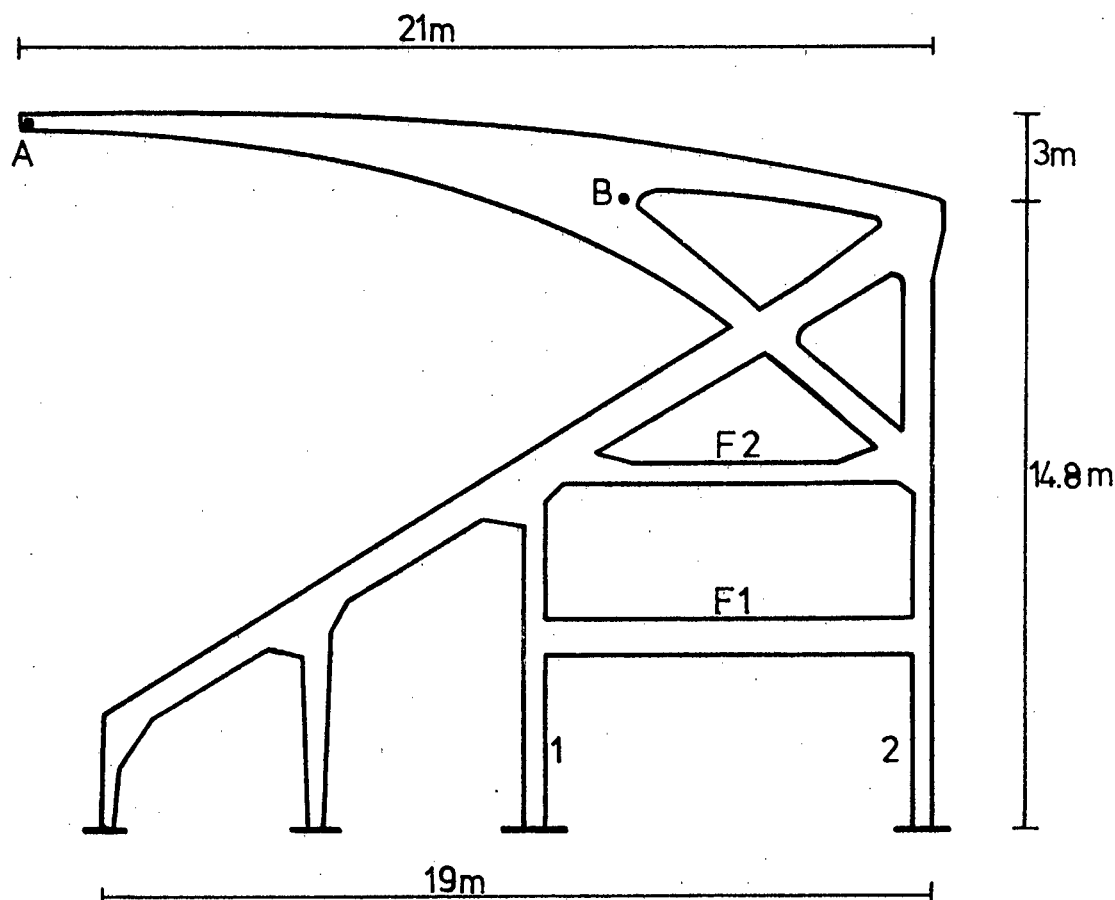
Generally, it can be said that equally accurate results are obtained using 5 NONPRI elements (analysis I) and using 24 2-noded elements (analysis IV). This ratio of 1 to 5 gives an idea of how accurate the non-prismatic formulation is.

In analysis II, where one single 3-noded element is used to model the centre span depth variation parabolically, reasonably accurate results are obtained for forces and moments (maximum error is 4%) but displacement results are not accurate (16% error). However displacements might not be an important factor for design in which case the approximation made in analysis II is quite suitable.

In general, it can be said that elements with the internal node located away from the centre of the element will yield poorer results than

elements with the internal node at the centre.

6.2.4 Grandstand Plane Frame Designed by Nervi [56]



LOADING (UNIFORMLY DISTRIBUTED) ON:

ROOF: 6 KN/m^2

FLOOR F1: 9 KN/m^2

SEATS: 12 KN/m^2

FLOOR F2: 9 KN/m^2

BACK WALL: 2.5 KN/m^2 (PRESSURE)

Figure 6.5 Nervi's grandstand - plane frame

Nervi's grandstand built in Florence in 1927, as shown in figure 6.5, is presented here simply to emphasize the advantages of the NONPRI element when dealing with more complicated structures. This grandstand was modelled for a realistic set of design loads simply with 22 NONPRI elements. This is taking exactly into account all the tapering and haunched members. As a comparison, it is stated that roughly 3 times as many 2-noded elements were required to achieve the

same set of results. It should be evident that the savings achieved using the NONPRI element are quite substantial - not only in computer time but also in input preparation and output interpretation time.

A few interesting results obtained from the NONPRI analysis are now given:

- Vertical deflection at A : 191 mm
- Largest bending moment of 1749 kNm occurs in the structure at point B. This is in fact where the frame cross-section is deepest.
- The resultant of all the loads coming down on columns 1 and 2 falls within the two columns. This eliminates the necessity of costly foundation anchorages. (as recognised by Nervi).

6.2.5 Three Dimensional Frame Highway Bridge Structure

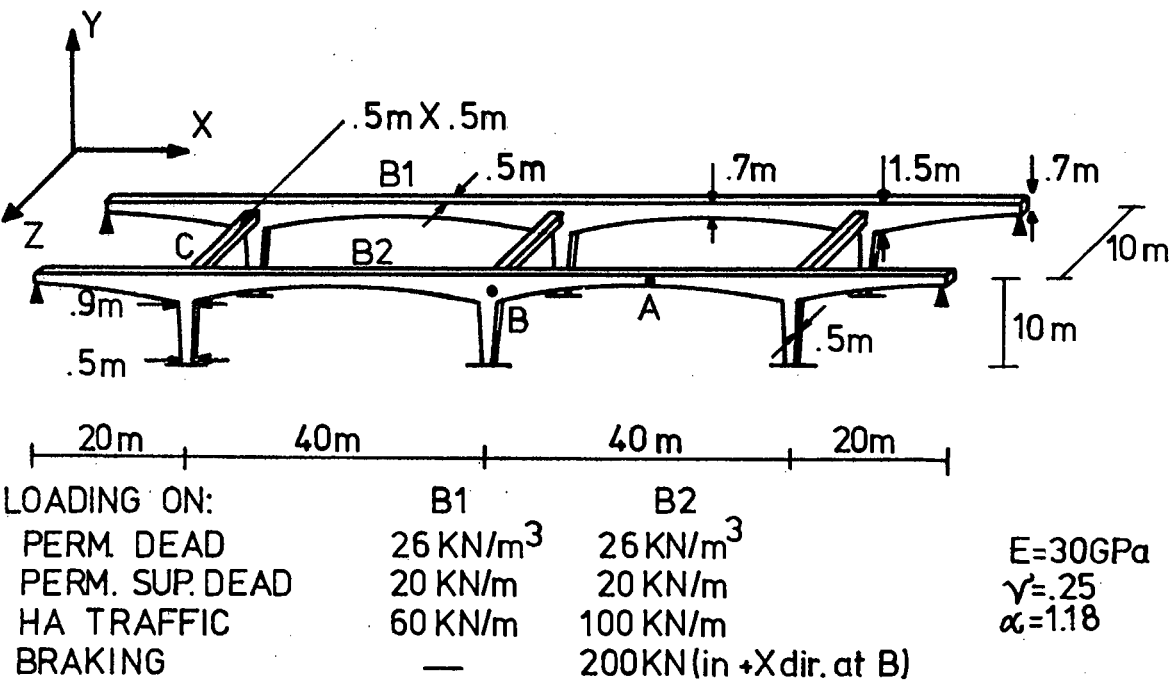


Figure 6.6 Highway bridge structure

The bridge frame of figure 6.6 is yet another example of the tremendous saving which the non-prismatic formulation can provide in getting a complete solution for the problem. This structure was modelled with only 17 NONPRI elements and as a result the analysis of four load cases required only 20 seconds of CPU time in a UNIVAC 1100/81 computer.

A few results obtained from the NONPRI analysis for the loading condition shown in the figure - permanent dead and superimposed dead loads, HA traffic live load and braking load - are given:

- Highest vertical (y) deflection of 77 mm occurs in the structure at point A.
- The horizontal braking load causes a relative horizontal (X) displacement between the two 4-span main beams of approximately 31 mm.
- Largest bending moment (about Z) of 18545 kNm occurs in the structure at point B.
- Torsional moment of 132 kNm occurs in cross beam C.

6.2.6 Execution Times for Multi-Storey Three Dimensional Frame

Recently, the designers of a multi-storey building to be erected in Cape Town, felt the need for a three dimensional analysis of the building's space frame podium structure.

This analysis, involving 184 elements, 300 nodes and 2 distinct loading conditions, was performed using NONPRI.

Without going into the details of the analysis, and simply to give an idea of the times taken by NONPRI to solve large problems, it is useful to mention the following run times for solution of the 2 load cases:

- 1 minute and 58 seconds CPU time for the analysis of the structure.
- 2 minutes and 16 seconds CPU time for the analysis including the interpolation option which gives 10 values of resultant forces and moments along each element.

The machine used was a UNIVAC 1100/81 computer

6.3 Dynamic Analyses

6.3.1 Cantilever Beam

The vibration analysis capabilities of NONPRI were used to calculate the natural frequencies and associated modes of vibration of the cantilever beam given in figure 6.7, in which the number of finite elements used was varied to study the convergence of the solution. Furthermore, different mass matrix evaluation schemes and the effect of rotatory inertia were also studied.

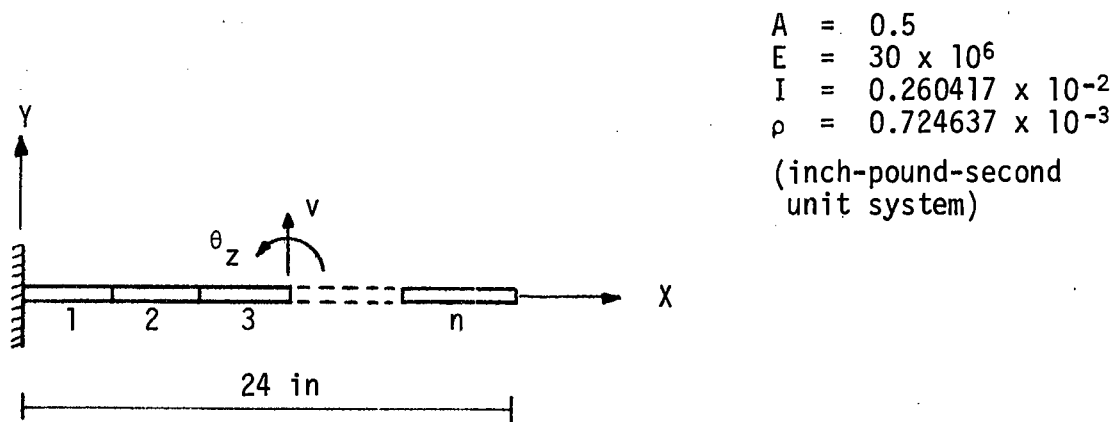


Figure 6.7 Cantilever beam with n-element idealisation

Each node of the cantilever is assumed to have 2 degrees of freedom, 1 transverse translation and 1 rotation. The results obtained are compared with those given by Gupta [46] where he included the effect of rotatory inertia in the analysis.

Table 6.4(a) displays the natural frequency results obtained for the analysis including rotatory inertia effects, while Table 6.4 (b) gives the results when effects due to rotatory inertia are neglected. Figure 6.8 shows the first five modes of vibration.

No. of Elements	Mass Matrix Calc. Scheme	Natural Frequencies (Rad./Sec.)				
		ω_1	ω_2	ω_3	ω_4	ω_5
1	E	72.9759	739.7966	-	-	-
	DC	91.2704	591.5065	-	-	-
	C	92.6419	1009.5258	-	-	-
2	E	84.7966	469.5056	1939.2406	6727.5934	-
	DC	89.7578	586.9163	1615.2488	6123.3427	-
	C	89.8417	599.5143	2490.6677	16452.7280	-
3	E	87.5200	516.9248	1356.7659	3846.9000	7285.8922
	DC	89.6851	565.3532	1680.9238	3160.3742	6796.9173
	C	89.8826	569.7348	1736.3234	4345.1660	13144.0510
4	E	88.4078	535.2837	1454.0840	2698.6188	6431.6749
	DC	89.5794	562.5882	1597.2938	3335.1577	5219.4327
	C	89.4696	564.2182	1626.2358	3465.0869	6861.8185
5	E	88.8426	544.3643	1494.5103	2863.3162	4493.1929
	DC	89.7587	561.9318	1581.3616	3154.4092	5552.9278
	C	89.7914	562.7014	1594.6933	3241.2459	5794.2415
Gupta [46] - 5 elements		89.6356	561.932	1577.96	3116.84	5170.29

Table 6.4(a) Rotatory Inertia Effects Included

No. of Elements	Mass Matrix Calc. Scheme	Natural Frequencies (Rad./Sec.)				
		ω_1	ω_2	ω_3	ω_4	ω_5
1	E	72.9757	740.0729	-	-	-
	DC	91.2704	591.5840	-	-	-
	C	92.6491	1009.7772	-	-	-
2	E	84.7966	469.5207	1940.8401	6762.8879	-
	DC	89.7578	587.0014	1615.7836	6140.8954	-
	C	89.8417	599.5967	2492.3346	16516.6110	-
3	E	87.5200	516.9751	1357.0365	3853.0215	7310.4736
	DC	89.6851	565.4275	1681.6403	3162.2999	6809.8541
	C	89.8826	569.8266	1737.0273	4349.4673	13188.7750
4	E	88.4078	535.3412	1454.4713	2699.6140	6448.6357
	DC	89.5794	562.6772	1597.8424	3338.0392	5224.5409
	C	89.4696	564.2686	1626.8312	3467.7013	6872.0024
5	E	88.8429	544.4445	1494.9737	2864.7305	4495.7006
	DC	89.7587	562.0543	1581.9254	3156.5741	5560.6356
	C	89.7914	562.7945	1595.2026	3243.4733	5801.2005

Table 6.4(b) Rotatory Inertia Effects Neglected

- E means an equal lumping of mass scheme
DC means a diagonalised consistent mass scheme
C means a consistent mass matrix scheme

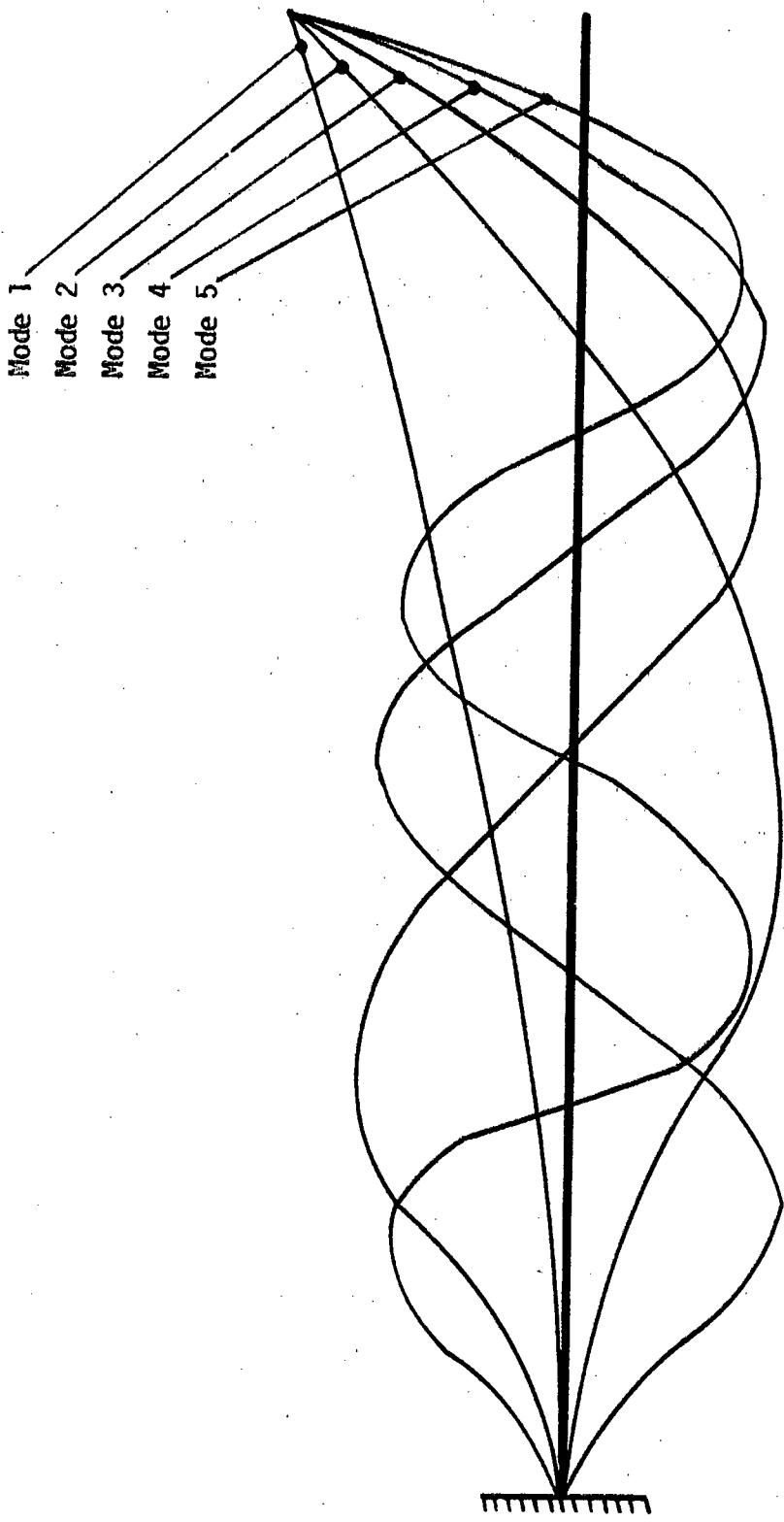


Figure 6.8 Cantilever beam and its first five mode shapes

The conclusions drawn from this study are:

- Convergence is observed in all cases.
- As expected, convergence becomes slower for the higher modes of vibration.
- Only 2 elements are needed to get an accurate answer for the first natural frequency.
- As stated in Chapter 4, generally, the diagonalised-consistent mass matrix scheme (special lumping based on the consistent matrix) yields the best results. The equal lumping mass matrix scheme gives the poorest results (underestimating the natural frequencies), while the consistent mass matrix scheme gives good results but always overestimating the natural frequencies.
- The effects of rotatory inertia, in this case, are insignificant.
- The results obtained by NONPRI compare very favourably with those given by Gupta [46].

6.3.2 Semicircular Arch

The semicircular arch given in figure 6.9 is the next example. The elastic and inertial properties of the arch are taken to be identical to the ones of the cantilever beam of 6.3.1.

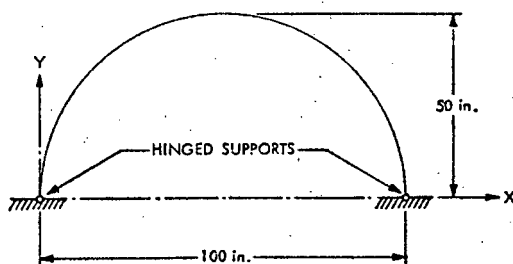


Figure 6.9 Semicircular arch

Ten straight line elements are used to approximate the arch structure. Again, different mass matrix evaluation schemes and the effect of rotatory inertia are investigated. Three degrees of freedom are considered at each node - 2 translations and 1 rotation. The results obtained are compared to those given by Gupta [46] where he included the effects of rotatory inertia in the analysis.

Table 6.5 gives the results obtained for the natural frequencies, while figure 6.10 shows the first four mode shapes.

Mass Matrix Calc. Scheme	Rotatory Inertia ?	Natural Frequencies (Rad./Sec.)						
		ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7
E	Yes	14.1275	41.8604	84.1512	137.5631	203.9628	280.9849	367.7706
DC	Yes	14.0866	41.7581	84.0610	137.8400	205.6953	285.6497	383.2029
C	Yes	14.1388	41.7987	84.3014	138.9017	209.0037	295.0407	402.3756
E	No	14.1178	41.8365	84.1356	137.5524	203.9747	281.0029	367.7884
DC	No	14.1063	41.7278	84.0423	137.8296	205.7033	286.6705	383.2251
C	No	13.9767	41.8199	84.3048	138.9093	209.0197	295.0498	402.3991
Gupta [46]	Yes	13.6831	41.6416	83.8944	137.145	203.500	281.061	371.301

Table 6.5

The following conclusions can be drawn:

- Again in this case the effects of rotatory inertia are very small.
- For the first 3 modes of vibration the diagonalised consistent mass matrix scheme gives the best results for the natural frequencies, while for the higher modes of vibration the equal lumping mass scheme gives the more accurate answers.
- Generally, the results obtained by NONPRI show very good agreement when compared with Gupta's results [46].
- Note that in examples 6.3.1 and 6.3.2 the effects of rotatory inertia were expected to be negligible since both structures are formed by thin beams and therefore the contribution due to rotating masses is very small.

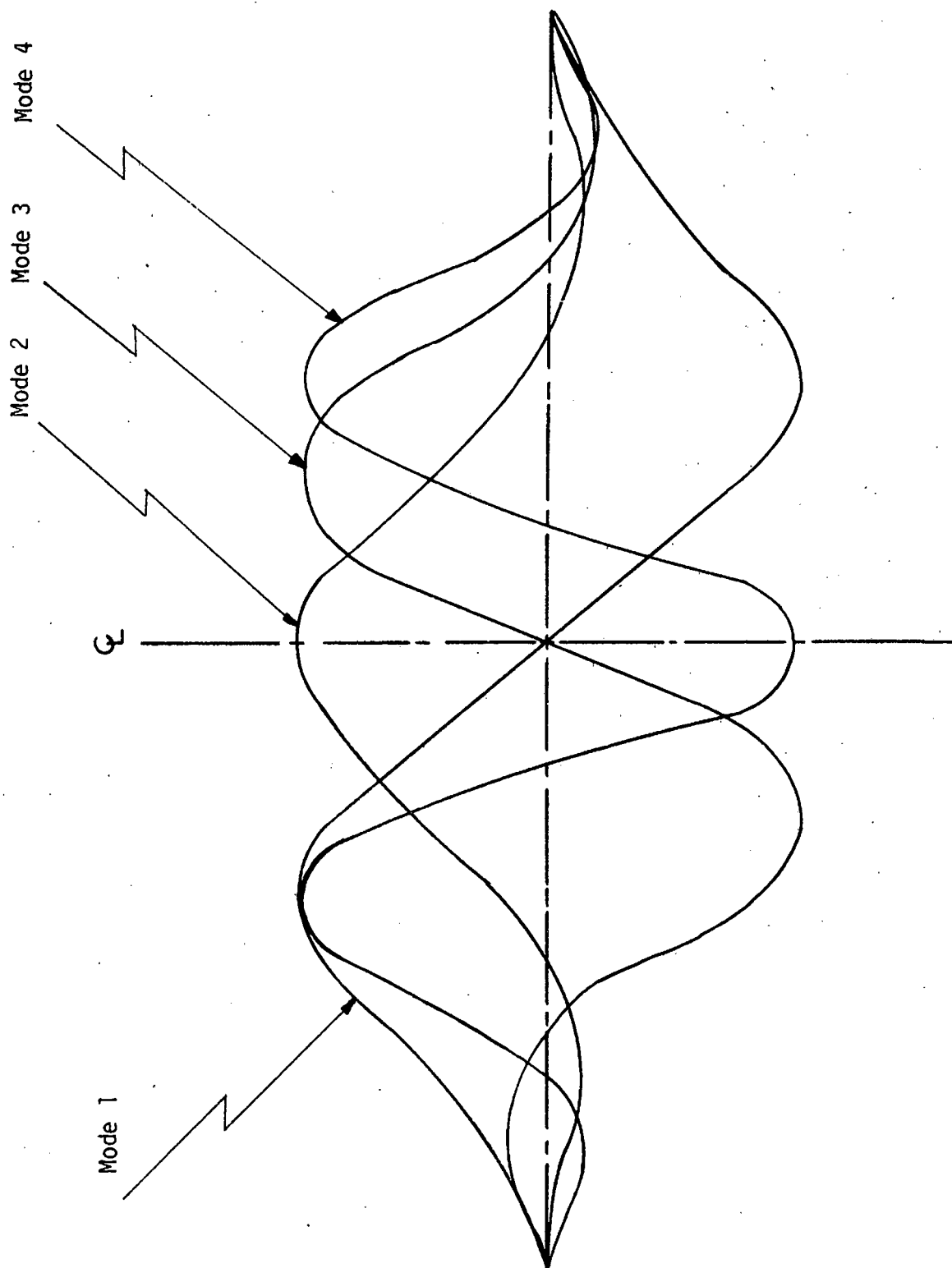
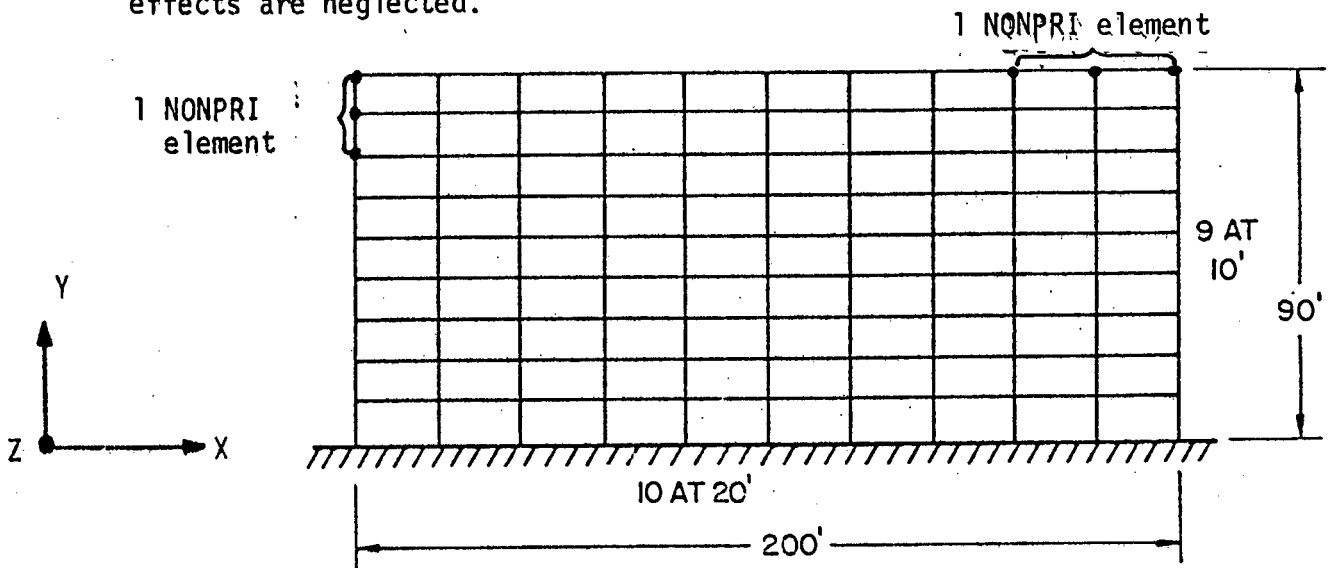


Figure 6.10 Semicircular arch modes of vibration.

6.3.3 Multi-storey, Multi-bay Plane Frame

The 9-storey, 10-bay plane frame given by Bathe and Wilson [49] is next analysed for the first six natural frequencies and associated mode shapes. The frame shown in figure 6.11 has 330 degrees of freedom (3 per node - 2 translations and 1 rotation) and the semi-bandwidth of the system is 62.

A diagonalised-consistent mass scheme is used and rotatory inertia effects are neglected.



YOUNG'S MODULUS = 432000, MASS DENSITY = 1.0
FOR ALL BEAMS AND COLUMNS $A_1 = 3.0$, $I_1 = I_2 = I_3 = 1.0$
UNITS: FT, KIPS

Figure 6.11 NONPRI model of Plane Frame with intercrossing elements.

The interesting thing about the NONPRI model for this frame is that, in order to save on elements and therefore size of the system to be solved, the elements are modelled as intercrossing. The result is that each element models 2 structural beams (or columns) in the frame as shown in figure 6.11.

The results obtained by NONPRI for the first six natural frequencies are given in table 6.6 together with the results offered by Bathe and Wilson [49]. Good agreement is achieved.

Source	Natural Frequencies (Rad./Sec.)					
	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6
NONPRI	.771795	2.36157	4.0113	5.930	6.357	6.469
Bathe et al [49]	.767816	2.35095	4.0728	5.951	6.423	8.001

Table 6.6

6.3.4 Unsymmetrical Space Frame

The unsymmetrical space frame (given by Leung [51]) of figure 6.12 consisting of 47 elements has been analysed. The frame has 426 degrees of freedom corresponding to 6 degrees of freedom per unrestrained node. All members are of aluminium having the following properties:

$$E = 69 \times 10^9 \text{ N/m}^2$$

$$G = 26 \times 10^9 \text{ N/m}^2$$

$$\rho = 2.7 \times 10^3 \text{ kg/m}^3$$

Cross-section as shown in figure 6.12

$$\alpha \text{ (shear factor)} = 1.2$$

A diagonalised-consistent mass scheme is adopted and rotatory inertia effects are included.

The first four natural frequencies calculated by NONPRI are given in table 6.7 together with the frequencies obtained by Leung [51]. Again the corresponding first four mode shapes are shown in figure 6.13.

Source	Natural Frequencies (Rad./Sec.)			
	ω_1	ω_2	ω_3	ω_4
NONPRI	5.138	5.911	7.503	20.004
Leung [51]	5.099	5.838	7.420	19.155

Table 6.7

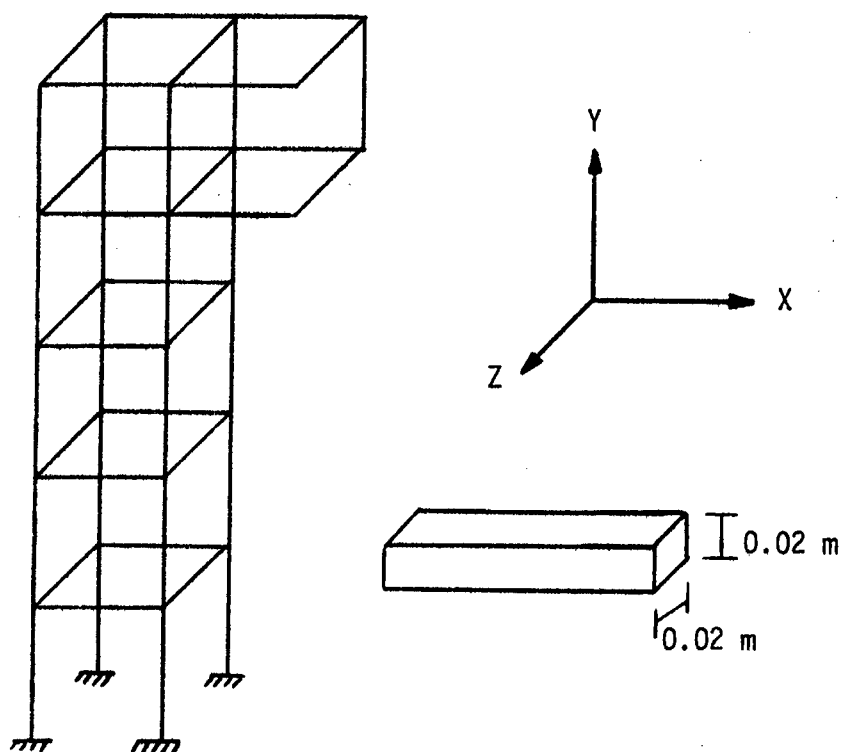
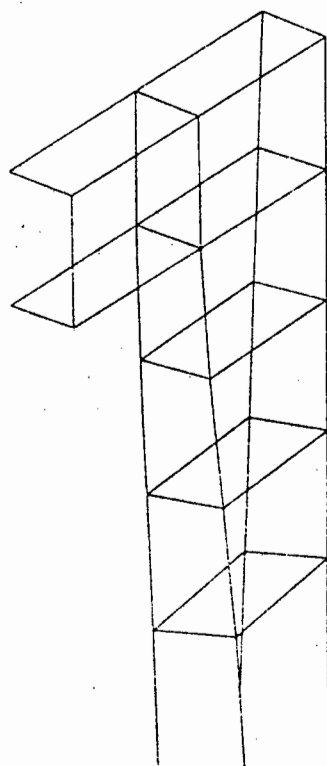
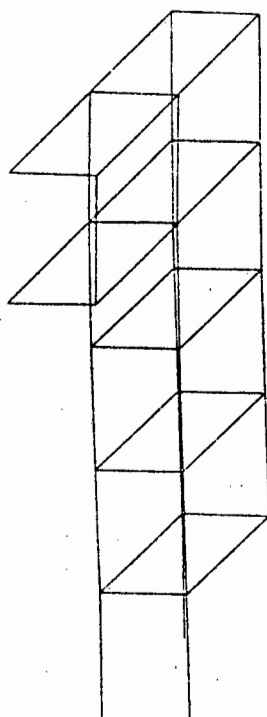


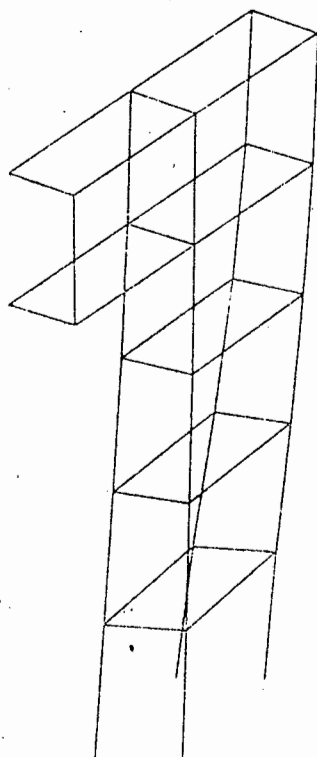
Figure 6.12 Unsymmetrical Space Frame and member size



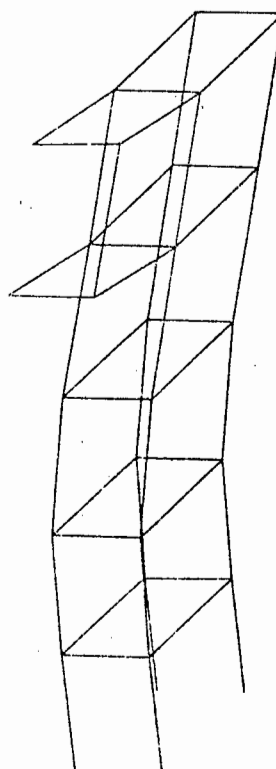
Mode 1



Mode 2



Mode 3



Mode 4

Figure 6.13 First four mode shapes for unsymmetrical space frame.

6.3.5 Tapered Cantilever

The tapering cantilever shown in figure 6.14 and suggested by Gallagher and Lee [6] was chosen to show the advantages of the present non-prismatic formulation in the context of dynamic analysis.

$$\begin{aligned} E &= 69 \times 10^9 \text{ N/m}^2 \\ G &= 26 \times 10^9 \text{ N/m}^2 \\ \rho &= 2.7 \times 10^3 \text{ kg/m}^3 \\ \alpha &= 1.2 \end{aligned}$$

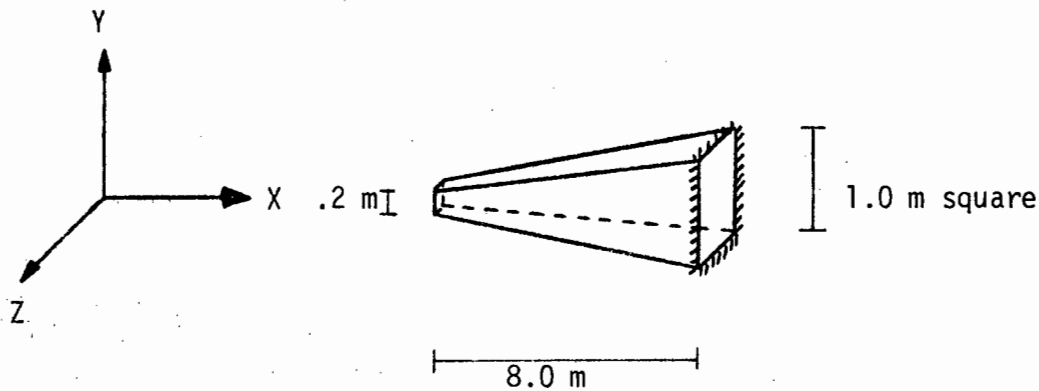


Figure 6.14 Tapering cantilever with square cross-section

As for example 6.2.2, two types of taper representation are used for comparison purposes, namely

1. The stepped representation using, this time, inertia averaging.
2. The NONPRI non-prismatic approximation.

The results obtained are given in figure 6.15 where a percentage error in natural frequencies vs number of elements plot is shown. The basis of comparison is drawn from results given by Wang [57] where a hyper-geometric series procedure is used to obtain solutions to the governing differential equation.

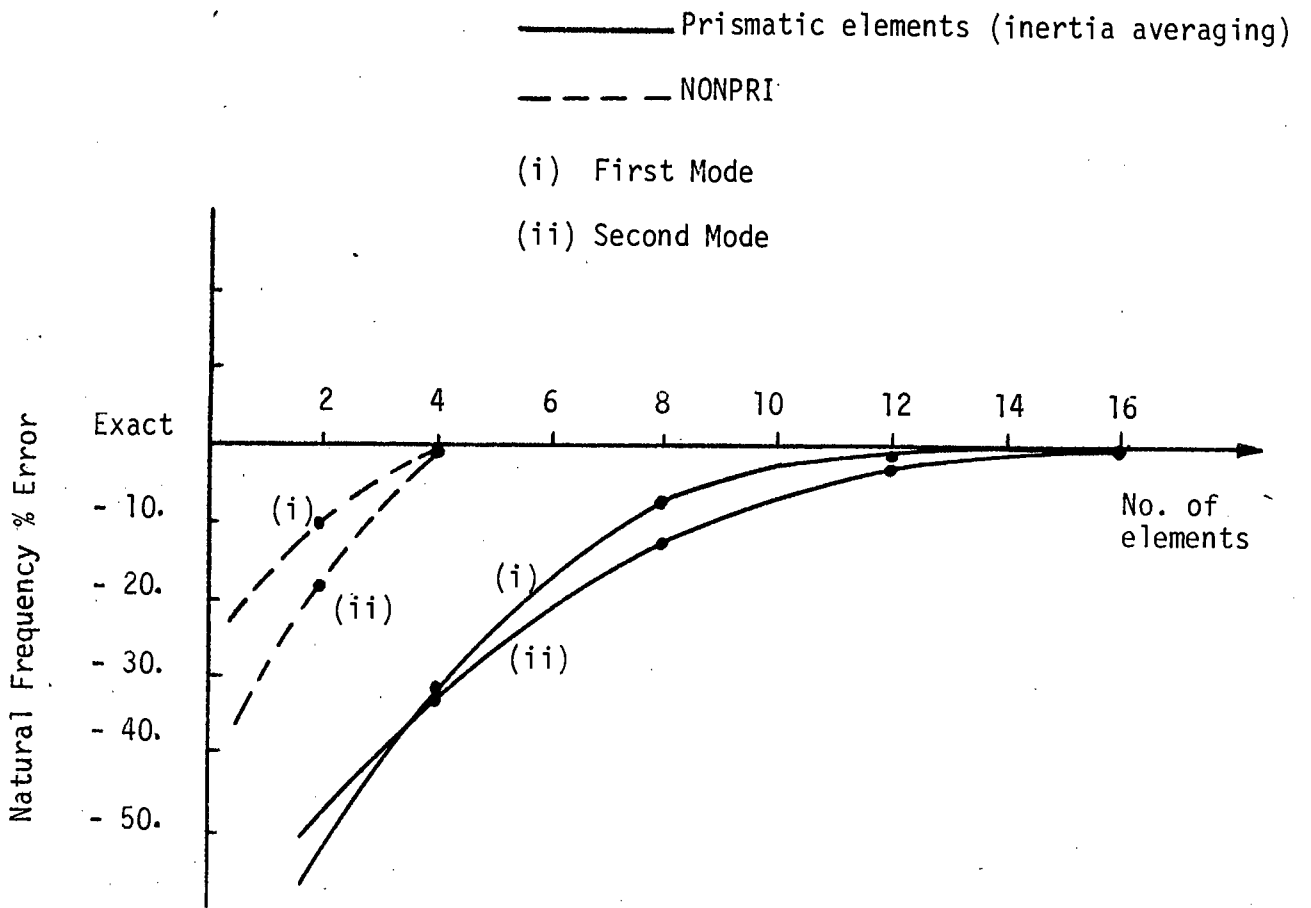


Figure 6.15 Natural frequency % error vs grid refinement

Again, the advantages of the present non-prismatic formulation become obvious when examining figure 6.15.

CHAPTER 7

CONCLUSIONS AND FURTHER DEVELOPMENTS

A non-prismatic three-dimensional beam finite element has been developed with results showing an improvement in the quality and economy of the analysis when comparing it with the available explicit 2-noded element formulation or the exponential function approximate formulation for non-prismatic beams.

Let us examine the main advantages gained from the presented formulation.

When one thinks of the conventional two-noded element in the context of static analysis it is immediately assumed that the displacement values and stress resultants are obtained at the two nodal points. This means that any discontinuities of, say, bending moment or shear force within the element cannot be 'picked up' and therefore wherever a discontinuity in the resulting quantities is expected to occur a new element must be started if an accurate solution is to be obtained. This leads, in many instances, to the use of an excessive number of elements in relation to the complexity of the structure. The existence of point loads or moments is an example of this condition. Further, if a coarse spacing of two-noded elements has been opted for, one might wish to calculate stress resultant values between the nodes. There is no automatic procedure for doing this when using a two-noded element.

The developed three-noded element is designed to overcome all these difficulties. This was achieved by the arbitrary definition of the geometrical position of the internal node within the element. It allows one to place the internal node wherever (within certain limits for acceptable accuracy) a point effect (load, moment or prescribed displacement) occurs and therefore 'pick up' any possible discontinuities of resultant quantities within the element. The fact that in

each half-element three values of each resultant quantity are sampled provides the facility of interpolating values anywhere along the element.

The allowed degeneration of the developed element to a truss/frame transition element and further down to a pure truss element has made the element very useful and powerful from the point of view of practical application.

When structures are composed of tapered members, the usual approach is to use the stepped representation of the taper. It has been shown in this work that the use of relatively few elements in the stepped representation of a tapered member can give a very low level of solution accuracy. This is not significant in the analysis of a beam with one or two spans where a refined 'grid' can be, and is, normally used. However, in frameworks consisting of non-prismatic members, a lot of care has to be taken in the 'grid' refinement if few elements are to be used for each tapered member.

Thus, it is desirable to formulate non-prismatic elements for the analysis of tapered members. A very practical and strong motivation for this approach has been the fact that dynamic analysis, due to its iterative nature, usually requires formidable computational effort and expense which is drastically reduced by establishing mass and stiffness matrices of acceptable accuracy with the least number of elements and therefore degrees of freedom.

The results obtained (some of which were presented in Chapter 6) lead to the belief that the present formulation can become a useful analysis tool for a great variety of structural applications.

Scope for further developments

The work carried out up to the stage of producing this document has opened many possibilities for further research and development.

Firstly, the ideas of the non-prismatic formulation can now be extended to develop a shell element with variable cross-section

characteristics to complement the already developed non-prismatic beam element. Secondly, the beam element could be further developed to include the capability of analysing curved members.

Thirdly, a transient response type of analysis can now be easily incorporated in the program. This would be an analysis by mode superposition using the natural frequencies and mode shapes obtained in the free vibration analysis already developed.

Fourthly, a more efficient solution method for the equations of vibration analysis in the case of large systems could be investigated. Here specific reference is made to the Subspace Iteration method developed by Bathe and Wilson.

Design orientated solution capabilities, namely incremental construction analysis and cable stayed structure analysis, could be incorporated in the developed program requiring no great effort.

Non-linear analysis capability is yet another line of development suggested for the present program.

Finally, as a long term project, it is thought that this work has laid the foundations of what could become a useful finite element structural analysis package - this would obviously require the enrichment of the element library, the inclusion of capabilities for various types of analysis not yet developed, and the addition of pre- and post-processing capabilities.

REFERENCES

1. ZIENKIEWICZ, O.C., "The Finite Element Method", 3rd Edition, McGraw Hill, London, 1977.
2. HINTON, E., OWEN, D.R.J., "Finite Element Programming", Academic Press, 1977.
3. DESAI, C.S., "Elementary Finite Element Method", Prentice Hall, 1979.
4. BATHE, K.J., WILSON, E.L., "Numerical Methods in Finite Element Analysis", Prentice Hall, 1976.
5. PAFEC, Ltd., "Pafec 75 - Program for Automatic Finite Element Calculations", Pafec Ltd., 1978.
6. GALLAGHER, R.H., LEE, C.H., "Matrix Dynamic and Instability Analysis with Non-uniform Elements", International Journal for Numerical Methods in Engineering, Vol. 2, 265-275, 1970.
7. MINDLIN, R.D., "Influence of Rotatory Inertia and Shear on Flexural Motions of Isotropic Elastic Plates", Journal of Applied Mechanics, 18, 31-38, 1951.
8. FRIED, I., "Numerical Integration in the Finite Element Method", Computers and Structures, Vol. 4, 921-932, 1974.
9. HUGHES, J.R., COHEN, M., HAROUN, M., "Reduced and Selective Integration Techniques in the Finite Element Analysis of Plates", Nuclear Engineering and Design, 46, 203-222, 1978.
10. ZIENKIEWICZ, O.C., TAYLOR, R.L., TOO, J.M., "Reduced Integration Technique in General Analysis of Plates and Shells", International Journal for Numerical Methods in Engineering, 3, 275-290, 1971.
11. KREYSZIG, E., "Advanced Engineering Mathematics", 4th Edition, Wiley, 1962.
12. AYRES, F.J.R., "Matrices", Schaum Publishing Company, New York, 1962.
13. KARDESTUNCER, H., "Elementary Analysis of Structures", McGraw-Hill, Kogakusha, 1974.
14. PICA, A., HINTON, E., "Transient and Pseudo-Transient Analysis of Mindlin Plates", International Journal for Numerical Methods in Engineering, Vol. 15, 189-208, 1980.
15. ODEN, J.T., "Finite Elements of Non-Linear Continua", McGraw-Hill, 1971.
16. HINTON, E., CAMPBELL, J., "Local and Global Smoothing of Discontinuous Finite Element Functions using a Least Square Method", International Journal for Numerical Methods in Engineering, Vol. 8, 461-480, 1974.

17. BARLOW, J., "Optimal Stress Locations in Finite Element Models", International Journal for Numerical Methods in Engineering, Vol. 10, 243-251, 1976.
18. HINTON, E., SCOTT, F.C., RICKETTS, R.E., "Local Least Squares Stress Smoothing for Parabolic Isoparametric Elements", International Journal for Numerical Methods in Engineering, Vol. 9, 235-256, 1975.
19. GERALD, C.F., "Applied Numerical Analysis", Addison Wesley, 1970.
20. RESENDE, L., DOYLE, W.S., "NONPRI - An Effective Non-Prismatic Three-Dimensional Beam Finite Element", Computers and Structures, to appear.
21. IRONS, B.M., "A Frontal Solution Program for Finite Element Analysis", International Journal for Numerical Methods in Engineering, Vol. 2, 5-32, 1970.
22. CHEUNG, Y.K., YEO, M.F., "A Practical Introduction to Finite Element Analysis", Pitman, 1979.
23. COLLINS, R.J., "A Modified Prefrontal Routine", International Journal for Numerical Methods in Engineering, Vol 11, 765,766, 1977.
24. MONDKAR, D.P., POWELL, G.H., "Towards Optimal In-core Equation Solving", Computers and Structures, Vol. 4, 531-548, 1974.
25. ABBAS, S.F., "Some Novel Applications of the Frontal Concept", International Journal for Numerical Methods in Engineering, Vol. 15, 519-536, 1980.
26. CEDOLIN, L., GALLAGHER, R., "A Frontal-Based Solver for Frequency Analysis", International Journal for Numerical Methods in Engineering, Vol. 12, 1659-1666, 1978.
27. YEO, M.F., "A More Efficient Front Solution: Allocating Assembly Locations by Longevity Considerations", International Journal for Numerical Methods in Engineering, Vol. 6, 570-574, 1973.
28. COLLINS, R.J., "Dynamic Destinations: A Method for Removing Redundant Operations in the Front Algorithm", International Journal for Numerical Methods in Engineering, Vol. 12, 1042-1043, 1977.
29. HOOD, P., "Frontal Solution Program for Unsymmetric Matrices", International Journal for Numerical Methods in Engineering, Vol. 10, 379-399, 1976.
30. GUYAN, R.J., "Reduction of Stiffness and Mass Matrices", AIAA Journal, Vol. 3, 380, 1965.
31. HINTON, E., BICANIC, N., "A Comparison of Lagrangian and Serendipity Mindlin Plate Elements for Free Vibration Analysis", Computers and Structures, Vol 10, 483-493, 1979.

32. KANAKA RAJU, K., HINTON, E., "Non-Linear Vibrations of thick Plates using Mindlin Plate Elements", International Journal for Numerical Methods in Engineering, Vol. 15, 249-257, 1980.
33. HINTON, E., ROCK, T., ZIENKIEWICZ, O.C., "A Note on Mass Lumping and Related Processes in the Finite Element Method", Earthquake Engineering and Structural Dynamics, Vol. 4, 245-249, 1976.
34. FRIED, I., MALKUS, D.S., "Finite Element Mass Matrix Lumping by Numerical Integration with no Convergence Rate Loss", International Journal of Solids and Structures, Vol. 11, 461-466, 1975.
35. MARTINS, R.A.F., OWEN, D.R.J., "Thin Plate Semiloof Element for Structural Analysis - Including Stability and Natural Vibrations", International Journal for Numerical Methods in Engineering, Vol. 12, 1667-1676, 1978.
36. CRANDALL, S., "Engineering Analysis", McGraw Hill, 1956.
37. WILKINSON, J.H., "The Algebraic Eigenvalue Problem", Clarendon Press, Oxford, 1965.
38. LANCZOS, C., "Applied Analysis", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1956.
39. KEY, S.W., BEISINGER, Z.E., in: Proceedings 3rd Conference on Matrix Methods in Structural Mechanics (Wright-Patterson Air Force Base, Ohio, 1971).
40. ARCHER, J.S., "Consistent Mass Matrix for Distributed Mass Systems", Journal Structural Division, ASCE, Vol. 89, ST4, 161-178, 1963.
41. CLOUGH, R.W., "Analysis of Structural Vibration and Response", in Recent Advances in Matrix Methods of Structural Analysis and Design (Ed. R.H. Gallagher, Y. Yamata and J.T. Oden), (First U.S.-Japan Seminar, 1969), Alabama Press, 1971.
42. WASHIZU, K., "Some Remarks on Basic Theory for Finite Element Method", in Recent Advances in Matrix Methods of Structural Analysis and Design, (Ed. R.H. Gallagher, Y. Yamata, J.T. Oden), (First U.S.-Japan Seminar, 1969), Alabama Press, 1971.
43. FUJINO, T., OHSAKA, K., "The Heat Conduction and Thermal Stress Analysis by the Finite Element Method", Proceedings 2nd Conference on Matrix Methods in Structural Analysis, Wright-Patterson Air Force Base, Ohio, 1968.
44. GUPTA, K.K., "Solution of Eigenvalue Problems by Sturm Sequence Method", International Journal for Numerical Methods in Engineering, Vol. 4, 379-404, 1972.
45. GUPTA, K.K., "Vibration of Frames and Other Structures with Banded Stiffness Matrix", International Journal for Numerical Methods in Engineering, Vol. 2, 221-228, 1970.

46. GUPTA, K.K., "Solution of Quadratic Matrix Equations for Free Vibration Analysis of Structures", International Journal for Numerical Methods in Engineering, Vol. 6, 129-135, 1973.
47. BATHE, K.J., WILSON, E.L., "Solution Methods for Eigenvalue Problems in Structural Mechanics", International Journal for Numerical Methods in Engineering, Vol. 6, 213-226, 1973.
48. BATHE, K.J., WILSON, E.L., "Eigensolution of Large Structural Systems with Small Bandwidth", ASCE Proceedings, Eng. Mech. Division Journal, Vol. 99, 467-479, 1973.
49. BATHE, K.J., WILSON, E.L., "Large Eigenvalue Problems in Dynamic Analysis", ASCE Proceedings, Eng. Mech. Division Journal, Vol. 98, 1471-1485, 1972.
50. IRONS, B., "Structural Eigenvalue Problems: Elimination of Unwanted Variables", AIAA Journal, Vol. 3, 961-962, 1965.
51. LEUNG, A.Y.T., "An Accurate Method of Dynamic Condensation in Structural Analysis", International Journal for Numerical Methods in Engineering, Vol. 12, 1705-1715, 1978.
52. UTKU, S., "Sturm Sequences or Law of Inertia of Quadratic Forms?", Computers and Structures, Vol. 3, 419-420, 1973.
53. BATHE, K.J., "ADINA - A Finite Element Program for Automatic Dynamic Incremental Non-Linear Analysis", Report 82448-1, Acoustics and Vibration Laboratory, Department of Mechanical Engineering, M.I.T., 1975.
54. BOLOURCHI, S., BATHE, K.J., "A Geometric and Material Nonlinear Three-Dimensional Beam Element", Report 82448-4, Acoustics and Vibration Laboratory, Department of Mechanical Engineering, M.I.T., 1977.
55. LIGHTFOOT, E., "Moment Distribution", E. and F.N. Spon, London, 1961.
56. NERVI, P.L., "Aesthetics and Technology in Building", Oxford University Press, London, 1966.
57. WANG, H.C., "Generalised Hypergeometric Function Solutions on the Transverse Vibration of a Class of Non-uniform Beams", Journal of Applied Mechanics, Vol. 36, 702-708, 1967.

APPENDIX A

THIRD NODE METHOD FOR SPACE TRANSFORMATIONS

As explained in Chapters 2 and 4, it is necessary to transform the element matrices (stiffness and mass) from the local coordinate space (x, y, z) to the global coordinate space (X, Y, Z) . This involves the computation of the direction cosines which constitute the element transformation matrix T^e .

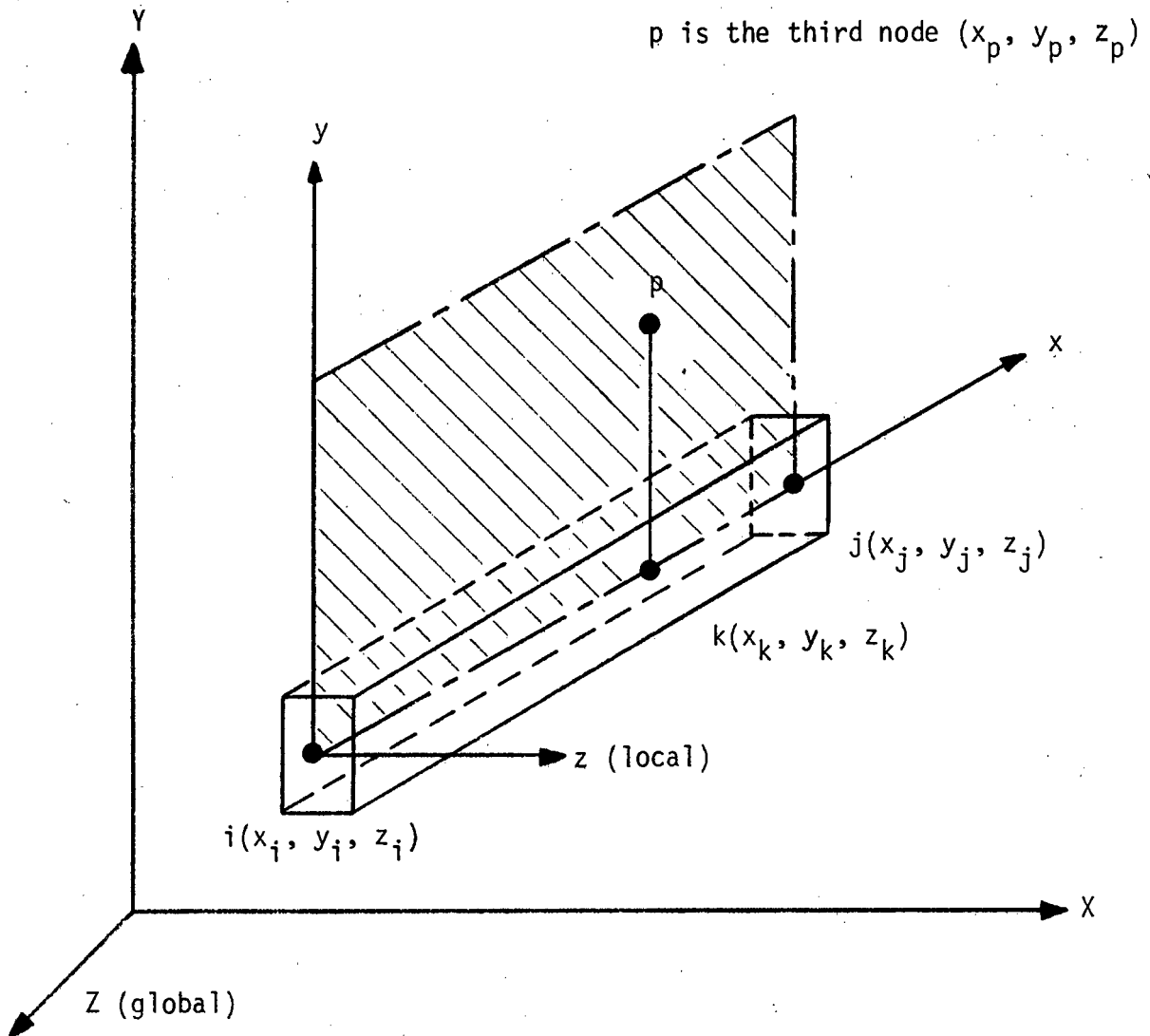


Figure A.1 The 3rd node method for arbitrarily orientated in space one-dimensional member.

There are 9 different direction cosines in the transformation matrix T^e .

$$DC = \begin{bmatrix} \cos (Xx) & \cos (Yx) & \cos (Zx) \\ \cos (Xy) & \cos (Yy) & \cos (Zy) \\ \cos (Xz) & \cos (Yz) & \cos (Zz) \end{bmatrix} \quad (A.1)$$

They, in fact, form the 3 x 3 array of direction cosines which in the program is called DC. The first row of direction cosines (local x-axis to global X, Y, Z axes - $\cos (Xx)$, $\cos (Yx)$, $\cos (Zx)$) is easily obtained as follows (figure A.1):

$$\begin{aligned} \cos (Xx) &= \frac{X_j - X_i}{L} \\ \cos (Yx) &= \frac{Y_j - Y_i}{L} \\ \cos (Zx) &= \frac{Z_j - Z_i}{L} \end{aligned} \quad (A.2)$$

$$\text{where } L = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2} \quad (A.3)$$

However, it becomes now necessary to specify a '3rd node' in order to define the position of the local y-axis since this can occupy any position in the full 360° range of orientations. This '3rd node' (point p on figure A.1) together with the element extremities i and j must define the element local x-y plane.

The evaluation of the second row of direction cosines (local y-axis to global X, Y, Z axes - $\cos (Xy)$, $\cos (Yy)$, $\cos (Zy)$) is the next step. For this, the coordinates of the projection of the '3rd node' p onto the element ij, i.e. the coordinates of point k have to be found. It is known that point k lies on the line defined by points i and j and therefore its coordinates are given by

$$\frac{X_k - X_i}{X_j - X_i} = \frac{Y_k - Y_i}{Y_j - Y_i} = \frac{Z_k - Z_i}{Z_j - Z_i} \quad (A.4)$$

Further, it is well known that the distance between two points in

a 3-D space is given as

$$\text{distance } \overline{pk} = \sqrt{(X_p - X_k)^2 + (Y_p - Y_k)^2 + (Z_p - Z_k)^2} \quad (\text{A.5})$$

and since k is the projection of p onto line ij , a minimum of distance \overline{pk} is implied. This occurs when

$$\frac{\partial \overline{pk}}{\partial X} = 0, \quad \frac{\partial \overline{pk}}{\partial Y} = 0, \quad \frac{\partial \overline{pk}}{\partial Z} = 0 \quad (\text{A.6})$$

Using these equalities, through some algebraic manipulation, it is possible to express the coordinates of point k in terms of known values and therefore possible to find the values of X_k , Y_k and Z_k .

The second row of direction cosines is now simply given by:

$$\begin{aligned} \cos (Xy) &= \frac{X_p - X_k}{\ell} \\ \cos (Yy) &= \frac{Y_p - Y_k}{\ell} \\ \cos (Zy) &= \frac{Z_p - Z_k}{\ell} \end{aligned} \quad (\text{A.7})$$

$$\text{where } \ell = \sqrt{(X_p - X_k)^2 + (Y_p - Y_k)^2 + (Z_p - Z_k)^2} \quad (\text{A.8})$$

To complete the calculation of the direction cosines, it is a matter of determining the third orthogonal axis direction cosines from the two known axes direction cosines. Through some involved matrix algebra and making use of orthogonality properties, it is possible to determine expressions for the last row of direction cosines in terms of the first two rows of direction cosines. These are:

$$\begin{aligned} \cos (Xz) &= \cos (Yx) \cos (Zy) - \cos (Yy) \cos (Zx) \\ \cos (Yz) &= \cos (Xy) \cos (Zx) - \cos (Xx) \cos (Zy) \\ \cos (Zz) &= \cos (Xx) \cos (Yy) - \cos (Xy) \cos (Yx) \end{aligned} \quad (\text{A.9})$$

With all the direction cosines now calculated it is just a matter of placing them in the right positions of the array to form the

transformation matrix T^e . The operations described in this appendix are carried out in subroutines DCOSSE and TMATSE of the program. As a major advantage of this '3rd node' method, it should be noted that it provides a quick and automatic means of defining the bending orientation of the beam/frame elements. Of course, the same '3rd node' can be associated with several beam/frame finite elements thus reducing the amount of input by the user.

APPENDIX B

NUMERICAL INTEGRATION

This appendix concerns the topic of numerical integration or quadrature repeatedly referred to in the main text of this work.

In the finite element method, one must evaluate certain integrals such as the stiffness integrals

$$K = \int B^T D B \, dx \quad (B.1)$$

the mass integrals

$$M = \int N^T \rho H N \, dx \quad (B.2)$$

and the load integrals

$$F = \int N q \, dx \quad (B.3)$$

In all these cases some sort of approximate numerical technique must be adopted due to the implicit nature of the non-prismatic formulation.

Basically, two different techniques can be used: Newton-Cotes and Gauss quadrature.

In NONPRI, a variation of Gauss quadrature is adopted - Gauss-Legendre quadrature - for its high accuracy and ease of computerization.

In general, the one-dimensional Gauss quadrature formula can be written

$$I_n = \int_{-1}^{+1} f(\xi) d\xi = \sum_{i=1}^n a_i f(\xi_i) \quad (B.4)$$

where a_i is a weighting factor
 ξ_i is the coordinate of the i th integration point
and n is the total number of integration points.

In Gauss-Legendre quadrature, specifically, instead of specifying the position of the sampling points *a priori* (as in the case of Newton-Cotes quadrature), one allows these to be located at points to be determined in order to achieve best accuracy. From equation (B.4) it is clear that for n sampling points, one has $2n$ unknowns (a_i and ξ_i) and hence a polynomial of order $2n - 1$ can be constructed and exactly integrated. The approximation error is thus of the order $O(\Delta^{2n})$. The simultaneous equations arising can be solved yielding an explicit solution in terms of Legendre polynomials - thus originating the technique called Gauss-Legendre quadrature.

Values of a_i and ξ_i for the first four Gauss-Legendre integration schemes are given in Table B.1 below.

n	i	ξ_i	a_i
1	I	0	2.00000 00000 00000
2	I	- 0.57735 02691 89626	1.00000 00000 00000
	II	+ 0.57735 02691 89626	1.00000 00000 00000
3	I	- 0.77459 66692 41483	0.55555 55555 55555
	II	0.00000 00000 00000	0.88888 88888 88889
	III	+ 0.77459 66692 41483	0.55555 55555 55555
4	I	- 0.86113 63115 94053	0.34785 48451 37454
	II	- 0.33998 10435 84856	0.65214 51548 62546
	III	+ 0.33998 10435 84856	0.65214 51548 62546
	IV	+ 0.86113 63115 94053	0.34785 48451 37454

Table B.1 Coordinates and weights for Gauss-Legendre Quadrature

Figure B.1 shows the sampling positions for the 2-point integration scheme.

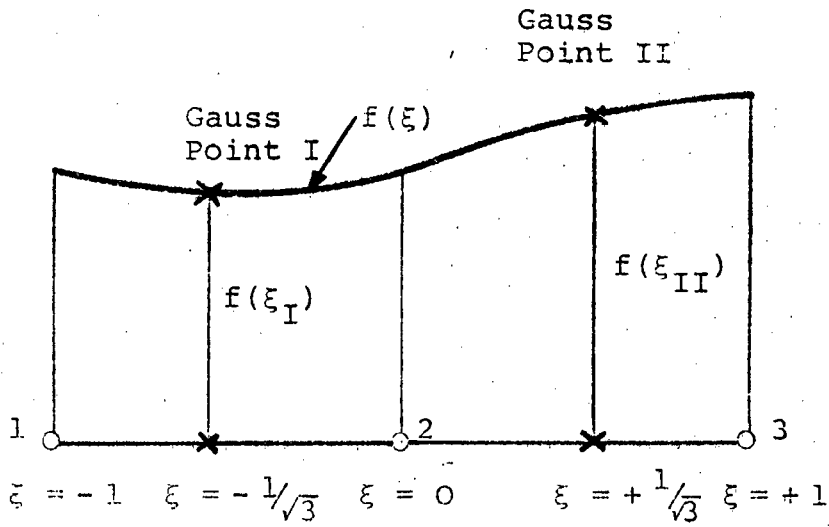


Figure B.1 2-point Gauss-Legendre integration for one-dimensional element.

Using the 2-point scheme, a typical integration for calculating consistent nodal loads is

$$\begin{aligned}
 F_i &= \int N_i q \, dx = \int N_i(\xi) q(\xi) \det J(\xi) d\xi \\
 &= a_I N_i(\xi_I) q(\xi_I) \det J(\xi_I) + a_{II} N_i(\xi_{II}) q(\xi_{II}) \det J(\xi_{II})
 \end{aligned}
 \tag{B.5}$$

APPENDIX C

NONPRI PROGRAM IMPLEMENTATION

C.1 Introduction

SADALE*NONPRI is a computer procedure for the Static And Dynamic Analysis of Linear Elastic systems of NON-PRIsmatic three dimensional beam finite elements.

It is divided into two main programs, the first performing the static analysis and the second performing the dynamic analysis. To make use of this suite of programs the user always has to assign the first program (the one dealing with static analysis) which will then control the sequence of execution of the required options and assigns the second program automatically in the case of a dynamic analysis being required.

C.2 Language and Machine

The program is written in UNIVAC ASCII FORTRAN and has been developed on a UNIVAC 1100/81 multi-processor system.

C.3 Program Structure

A modular approach is adopted in the organisation of NONPRI where the basic finite element steps are performed by primary subroutines which then rely on auxiliary subroutines to carry out secondary operations. The two main programs control the order of calling the primary routines as shown in figures C.1 and C.2.

C.4 Index of Routines

A list of the routines forming the program NONPRI is next presented together with a brief explanation of the functions they perform.

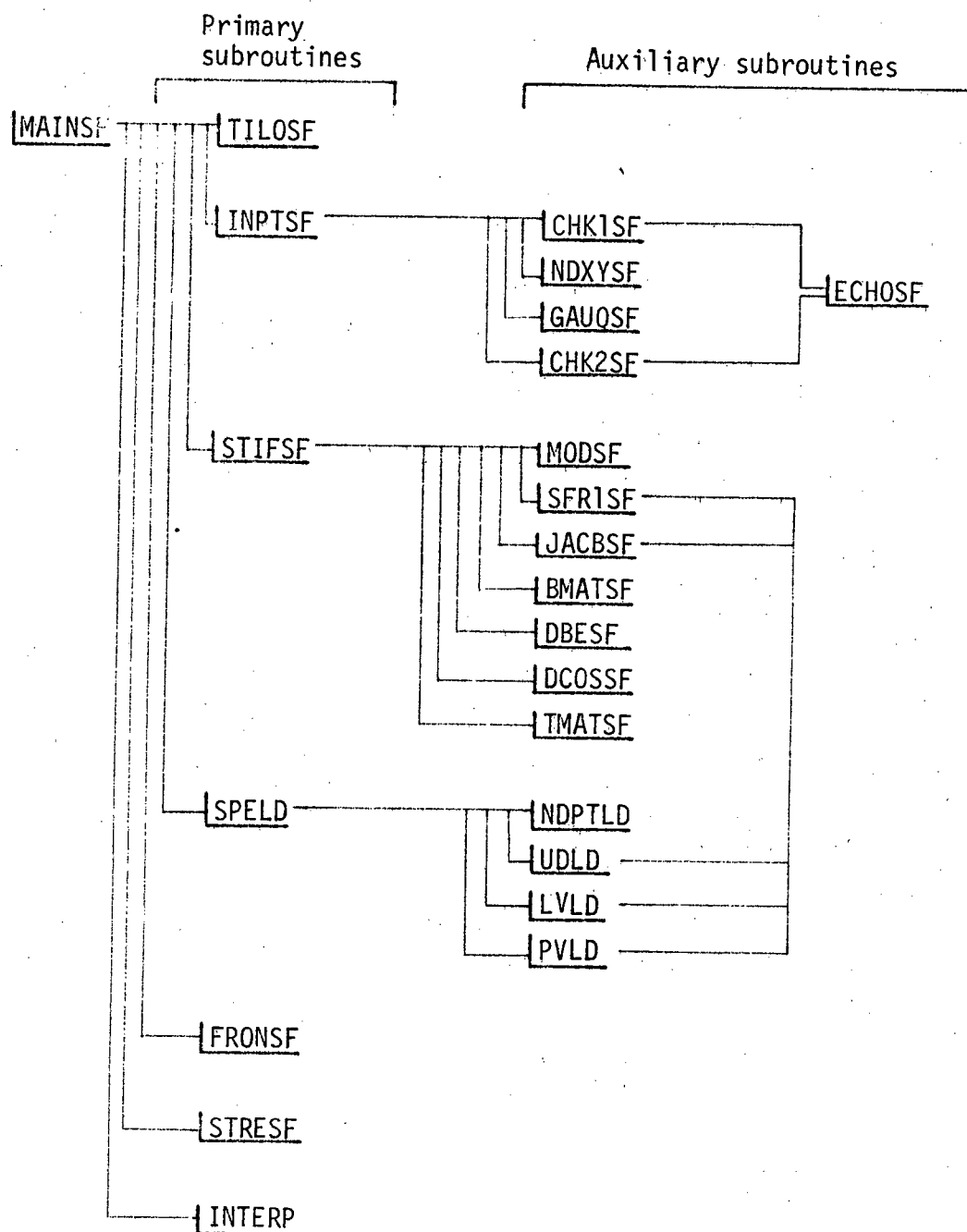


Figure C.1 Static analysis program structure

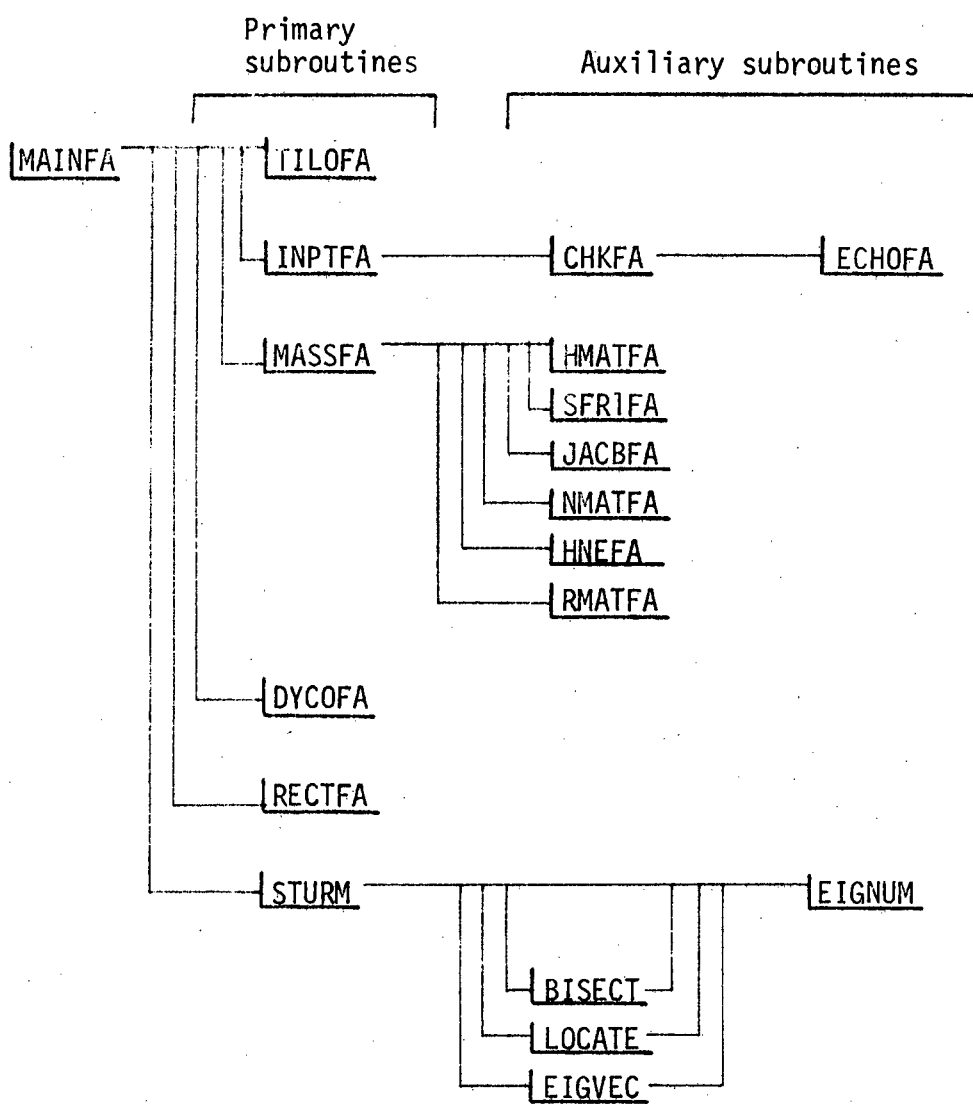


Figure C.2 Dynamic analysis program structure.

- STATIC ANALYSIS LIBRARY OF ROUTINES

- MAINSF: Master routine: controls the order of calling the primary routines according to the options exercised by the user.
- TILOSF: Print-out title block for static analysis.
- INPTSF: Input of control and structure data for static analysis. Controls calling of checking and data generation routines in static analysis.
- CHK1SF: Control data checking and diagnostics.
- NDXYSF: Generation of internal node coordinate data.
- GAUQSF: Setting up of sampling points and weighting factors for numerical integration.
- CHK2SF: Static analysis structure data checking and printing of any diagnostics/job abortion.
- ECHOSF: Reading and echoing by line printer of post-disaster data.
- STISF: Formulation of stiffness, stress and resultant force/moment matrices.
- MODSF : Setting up of stress/strain relationship D matrix at the Gauss sampling points.
- SFR1SF: Computation of shape functions and their local derivatives.
- JACBSF: Computation of Jacobian operator, its determinant, its inverse and the cartesian derivatives of the shape functions.
- BMATSF: Setting up of strain/displacement relationship B matrix.
- DBESF : Computation of matrix product D.B for stress calculations.
- DCOSSF: Calculation of direction cosines for space frame transformations.
- TMATSF: Setting up of element transformation matrix T^e .
- SPELD : Input and control of all load types. Controls calling of routines to handle all types of loads.
- NDPTLD: Generation of equivalent nodal loads for input point concentrated loads.

- UDLD : Generation of equivalent nodal loads for input uniformly distributed loads.
- LVLD : Generation of equivalent nodal loads for input linearly varying distributed loads.
- PVLD : Generation of equivalent nodal loads for input parabolically varying distributed loads.
- FRONSF: Solution of static equilibrium equations by the frontal method.
- STRESF: Calculation of all force and moment results from the nodal displacements.
- INTERP: Interpolation of force and moment resultants at half-element fifth points using Lagrangian interpolation polynomials.

- DYNAMIC ANALYSIS LIBRARY OF ROUTINES

- MAINFA: Main controlling routine for dynamic analysis.
- TILOFA: Print-out title block for dynamic analysis.
- INPTFA: Input of additional data for dynamic analysis.
- CHKFA : Dynamic analysis additional data checking and printing of any diagnostics/job abortion.
- ECHOFA: Reading and echoing by line printer of post-disaster data.
- MASSFA: Formulation of the different mass matrix schemes.
- HMATFA: Setting up of cross-sectional properties matrix H at the Gauss sampling points.
- SFR1FA: Computation of shape functions.
- JACBFA: Computation of Jacobian operator and its determinant.
- NMATFA: Setting up of shape functions matrix N.
- HNEFA : Computation of matrix product H.N.
- RMATFA: Setting up of mass density matrix ρ .
- DYCOFA: Basic condensation of unwanted global degrees of freedom and restrained degrees of freedom for dynamic analysis solution.

- RECTFA: To compact stiffness and mass matrices into skew rectangular form for dynamic analysis solution.
- STURM : Main driving routine for Sturm sequence based method of solution of vibration problems.
- EIGNUM: To calculate the number of eigenvalues within a specified interval.
- BISECT: To isolate each individual eigenvalue by means of bisection techniques.
- LOCATE: To locate and determine to a specified accuracy the value of each isolated eigenvalue.
- EIGVEC: To determine the corresponding eigenvector by inverse iteration techniques.

C.5 Data Reading, Generation, Checking and Diagnostics

Data, in NONPRI, is read in four different main sections: firstly, control data is read in INPTSF; secondly, structure data is read also in INPTSF; thirdly, loading data which specifically relates to static analysis is read in SPELD; and lastly, data relating to dynamic analysis is read in INPTFA. All data is read and organised in free format.

A small degree of data generation is included in NONPRI and this relates to the generation of internal node coordinates if they happen to be located at the centre of the element. This operation is performed by NDXYSF. Data checking and error diagnosing are items which have become essential in any program (of structural analysis or otherwise) if it is to be an efficient one. Therefore, attention to this topic has been given in the development of NONPRI and several routines have been included to perform these functions.

As soon as the control parameters are entered in INPTSF, CHK1SF is called to check them. If any errors are detected, subroutine ECHOSF is used to echo the remainder of the problem input data before the job is terminated. If no errors are detected in the control parameters, then the geometric data, element data, boundary conditions,

beam properties and material properties are assimilated by INPTSF and subsequently criticized by CHK2SF. The same procedure as above is followed if any errors are detected. The loading data entered in SPELD is checked as soon as it is read. The data relating to dynamic analysis entered in INPTFA is checked by CHKFA and as previously, if any errors are detected, routine ECHOFA is used to echo the post-disaster data. Up to this stage, only checking of input data has been discussed. However, NONPRI includes numerous other checks related to the operations it performs and some of these are: Jacobian determinant checks in routine JACBSF and JACBFA, definition of element local axes checks in routine DCOSF, maximum size of frontwidth checks in CHK2SF, maximum bandwidth size checks in routine RECTFA, and checks for positive-definiteness of stiffness and mass matrices in routine STURM.

Basically, two kinds of checks are performed: if a fatal error has been detected an error message is printed before the job is aborted; on the other hand, if a non-fatal error is detected a warning is printed and the run continues. Furthermore, for any error (fatal or non-fatal) encountered, a brief but fully explanatory diagnostic is printed for the benefit of the user.

C.6 Generation of Element Matrices

The element matrices are generated from the nodal coordinates, topologies, cross-sectional properties and material properties. The stiffness and mass matrices are first assembled in the element local set of axes and then transformed to the global set of axes by using the element transformation matrices obtained from the direction cosines. The loading matrices are directly assembled in the global set of axes.

C.7 Element Data Handling and Transfer

The stiffness, stress and force/moment resultant matrices are generated and written onto disc file element by element. Similarly, the transformation and loading matrices are written onto disc file once they have been generated. These matrices are read from disc file whenever they are required, for instance, during the solution the stiffness and

loading matrices are required and during the calculation of force/moment resultants the stress, force/moment resultants and transformation matrices are required.

The above operations take place in the static analysis program of NONPRI. However, if a dynamic analysis is required some of this element data has to be transferred from the static analysis program to the dynamic analysis program. This is the case with the element stiffness and transformation matrices which are written onto permanent files which are then read once the dynamic analysis program is initiated. The reason for using permanent files for this transfer is that the user might not wish to perform the dynamic analysis immediately after the static analysis, in which case the relevant element data from the static analysis is kept on permanent file. This information is automatically retrieved once the dynamic analysis program has been initiated.

In the course of the dynamic analysis program the mass matrix is written onto disc file element by element and (as for the stiffness matrix in static analysis) is read from the same disc file when it is needed for the dynamic equations solution.

C.8 Solution Routines

A frontal technique is used in NONPRI for the solution of the equilibrium equations of static analysis. It is capable of handling prescribed displacements/rotations as well as applied loads/couples. Furthermore, a multiple load case resolution facility has been included in NONPRI. At the termination of the frontal procedure, the user is provided with a list of nodal displacements and a list of reactions at restrained nodal points. The frontal procedure makes use of disc files in order to obtain the required elemental data as described in section C.7. The potential of the frontal solver, as far as size of problem it is capable of handling is concerned, is virtually unlimited, since it is an out-of-core procedure.

A banded type solver with reduced storage requirements is used for the solution of the equations of free vibration in the dynamic analysis program. The reduced storage requirements are obtained through the

use of skew rectangular compacted arrays. Being an in-core solver, the banded technique has limitations from the point of view of the size of problem that can be handled. However, believing that the iterative nature of the solution warrants an in-core solver, it has been decided to stick to it for as long as it is possible. It should be added that out-of-core procedures for dynamic analysis are being investigated at the time of writing this document. At the termination of the vibration solution the user is provided with the structural system natural frequencies and associated mode shapes.

C.9 Selective Output

Numerous output options have been built into NONPRI so as to allow the user to select exactly the output he requires and thus avoid the problem of having to search through unwanted output information. These options are detailed in the NONPRI program user manual.

C.10 Core Space Management

Overlay techniques have been extensively used throughout the development of NONPRI. This has resulted in considerable savings in core space requirements, thus making easier further developments of NONPRI. Also, all the routines constituting NONPRI have been compiled and mapped to create an absolute version of the program, previous to execution by the user. Therefore, only efforts relating to the actual solution of the problem are made at the time of execution.

APPENDIX D

NONPRI PROGRAM USER MANUAL

At the time of writing this document, the input procedure of NONPRI is undergoing major revisions and improvements. Therefore, it is thought sensible not to include here a program user manual which is out of date. A new version of the NONPRI program user manual will be available as a separate document in the near future.

APPENDIX E

SAMPLE OUTPUT FROM NONPRI

The computer printed output for the square cross-section tapered cantilever of example 6.3.5 is given in this appendix to illustrate the information obtained when using NONPRI.

The run here presented refers to the case when four non-prismatic NONPRI elements were used and a static as well as a dynamic analysis are performed.

The input for this run is attached at the end of the appendix.

USER WILL BE PROVIDED THE FULL ANALYSIS OUTPUT.

NONPRI IS A PROGRAM FOR THE STATIC LINEAR ELASTIC ANALYSIS OF THREE-DIMENSIONAL NONPRISMATIC BEAM ELEMENTS. THE FINITE ELEMENT DISPLACEMENT METHOD IS USED AND SHEAR DEFORMATION AS WELL AS BENDING, AXIAL AND TORSIONAL DEFORMATIONS ARE TAKEN INTO ACCOUNT. THE THREE NODED BEAM ELEMENTS CAN BE SPECIFIED TO HAVE THE INTERNAL NODE AWAY FROM THE CENTRE OF THE ELEMENT. THE THREE NODED BEAM ELEMENTS CAN BE SPECIFIED TO HAVE A DIFFERENT NUMBER OF ACTIVE DEGREES OF FREEDOM AT EACH NODE. A FRONTAL TECHNIQUE IS USED FOR THE SOLUTION OF THE SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS OF STATIC ANALYSIS. A RESOLUTION FACILITY FOR MULTIPLE LOAD CASES IS INCORPORATED IN THE FRONTAL SOLUTION. POINT FORCES AND COUPLES CAN BE SPECIFIED AND EXACTLY SOLVED PARABOLICALLY VARYING FORCES AND COUPLES CAN BE SPECIFIED AND EXACTLY SOLVED. INTERPOLATION FACILITIES ARE PROVIDED TO OBTAIN A MORE COMPLETE SOLUTION.

A large, stylized letter 'Z' composed of 'x' marks, with a smaller 'Z' made of 'z' marks above it.

GALLAGHER'S SQUARE SECTION TAPERING CANTILEVER - FOUR NONPRI

STATIC AND FREQUENCY ANALYSES TO BE PERFORMED!

NO. OF STRUCTURAL NODES= 9
NO. OF NON-STRUCTURAL NODES= 1
NO. OF ELEMENTS= 4
ORDER OF INTEGRATION FORMULA= 2

Table with 4 columns: ELEMENT, MATERIAL, TOPOLOGY, PROPERTY, 3RD. NODE. It lists values for elements 1 through 4.

Table with 4 columns: NODAL POINT COORDINATES (METRES), X, Y, Z. It lists coordinates for nodes 1 through 10.

RESTRAINED NODES

FIXITY CODE:
1 INDICATES D.O.F. IS RESTRAINED
0 INDICATES D.O.F. IS UNRESTRAINED

Table with 2 columns: NODE FIXITY, PRESCRIBED VALUES (M AND RAD.). It shows fixity values for nodes 1 through 10.

SECTION PROPERTIES (M UNITS)

Table with 5 columns: NUMBER, A, IYY, IZZ, A/ALPHA. It lists section properties for elements 1 through 9.

MATERIAL PROPERTIES (M, N AND KG UNITS)

Table with 4 columns: NUMBER, E, G, R0. It lists material properties for element 1.

MAX. FRONTWIDTH ENCOUNTERED= 18

DO YOU WISH TO ENTER LOADS TYPE L,
OR FORCES AND MOMENTS TYPE F.
L

LOADS TO BE ENTERED.

LOAD GENERATION ROUTINE ENTERED!

DO YOU WISH TO SPECIFY YOUR LOADS TYPE S,
OR LET ME GENERATE BRIDGE CODE LOADS TYPE C.

LOADS TO BE SPECIFIED.

ENTER THE NO. OF IMPOSED LOAD CASES!

NO. OF IMPOSED LOAD CASES= 1
TOTAL NO. OF LOAD CASES= 1

ELEMENT	NODE	X-LOAD	X-COUPLE	Z-LOAD	Y-COUPLE	Y-LOAD	Z-COUPLE
1	1	.0000	.0000	.0000	.0000	.0000	.0000
	2	.0000	.0000	.0000	.0000	.0000	.0000
	3	.0000	.0000	.0000	.0000	.0000	.0000
2	1	.0000	.0000	.0000	.0000	.0000	.0000
	2	.0000	.0000	.0000	.0000	.0000	.0000
	3	.0000	.0000	.0000	.0000	.0000	.0000
3	1	.0000	.0000	.0000	.0000	.0000	.0000
	2	.0000	.0000	.0000	.0000	.0000	.0000
	3	.0000	.0000	.0000	.0000	.0000	.0000
4	1	.0000	.0000	.0000	.0000	.0000	.0000
	2	.0000	.0000	.0000	.0000	.0000	.0000
	3	.0000	.0000	.0000	.0000	.0000	.0000
		-1000.	.

*** FRONTAL SOLUTION ROUTINE ENTERED!***

LOAD CASE NO. 1 : POINT LOAD AT TIP(-1000)

DISPLACEMENTS (IN GLOBAL AXES SYSTEM)-M & RAD UNITS

NODE	X-DISPL.	X-ROTAT.	Z-DISPL.	Y-ROTAT.	Y-DISPL.	Z-ROTAT.
1	.00000	.00000	.00000	.00000	.00000	.00000
2	.00000	.00000	.00000	.00000	.80754-006	.16080-005
3	.00000	.00000	.00000	.00000	.35224-005	.37892-005
4	.00000	.00000	.00000	.00000	.87979-005	.68300-005
5	.00000	.00000	.00000	.00000	.17841-004	.11270-004
6	.00000	.00000	.00000	.00000	.32266-004	.17973-004
7	.00000	.00000	.00000	.00000	.55573-004	.28880-004
8	.00000	.00000	.00000	.00000	.92660-004	.45700-004
9	.00000	.00000	.00000	.00000	.15049-003	.69621-004

REACTIONS (IN GLOBAL AXES SYSTEM)-M & N UNITS

NODE	X-FORCE.	X-MOMENT	Z-FORCE.	Y-MOMENT	Y-FORCE.	Z-MOMENT
1	.00000	.00000	.00000	.00000	.99999+003	.80000+004

*** STRESSING ROUTINE ENTERED!***

KEY FOR <NODE> COLUMN IN <FORCES AND MOMENTS> TABLE:
XL MEANS VALUE HAS BEEN CALCULATED JUST LEFT OF INTERNAL NODE X
XR MEANS VALUE HAS BEEN CALCULATED JUST RIGHT OF INTERNAL NODE X

LOAD CASE NO. 1 : POINT LOAD AT TIP(-1000)

FORCES AND MOMENTS (IN LOCAL AXES SYSTEM)-M AND N UNITS

NODE	X-COORD	Y-COORD	Z-COORD	AXIAL IN X	TORS MOM-X	SHEAR IN Z	BEND MOM-Y	SHEAR IN Y	BEND MOM-Z
ELEMENT NO. 1									
1	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.80000+004
2L	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.70000+004
2R	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.70000+004
3	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.60000+004
1				.00000	.00000	.00000	.00000	.99999+003	-.75773+004
									-.64226+004
ELEMENT NO. 2									
3	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.60000+004
4L	3.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.50000+004
4R	3.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.50000+004
5	4.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.40000+004
2				.00000	.00000	.00000	.00000	.99999+003	-.55773+004
									-.44227+004
ELEMENT NO. 3									
5	4.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.40000+004
6L	5.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.30000+004
6R	5.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.30000+004
7	6.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.20000+004
3				.00000	.00000	.00000	.00000	.99999+003	-.35774+004
									-.24227+004
ELEMENT NO. 4									
7	6.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.20000+004
8L	7.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.10000+004
8R	7.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.10000+004
9	8.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-.10000+004
4				.00000	.00000	.00000	.00000	.99999+003	-.14648+002
									-.15774+004
									-.42265+003

DO YOU WISH TO CALCULATE FORCES AND MOMENTS OFF THE NODES?

INTERPOLATION ROUTINE ENTERED!

VALUES AT SUBELEMENT FIFTH POINTS TO BE INTERPOLATED USING LAGRANGIAN POLYNOMIAL

KEY FOR <NODE>	COLUMN IN <FORCES AND MOMENTS>	TABLE:	OF	ELEMENT	X
X.1	IS AT 1/5 OF THE WAY BETWEEN 1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	X
X.2	IS AT 2/5 OF THE WAY BETWEEN 1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	X
X.3	IS AT 3/5 OF THE WAY BETWEEN 1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	X
X.4	IS AT 4/5 OF THE WAY BETWEEN 1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	1ST. & 2ND.	X
X.5	IS AT 1/5 OF THE WAY BETWEEN 2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	X
X.6	IS AT 2/5 OF THE WAY BETWEEN 2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	X
X.7	IS AT 3/5 OF THE WAY BETWEEN 2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	X
X.8	IS AT 4/5 OF THE WAY BETWEEN 2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	2ND. & 3RD.	X

FORCES AND MOMENTS (IN LOCAL AXES SYSTEM)-M AND N UNITS

NODE	X-COORD	Y-COORD	Z-COORD	AXIAL IN X	TORS MOM-X	SHEAR IN Z	BEND MOM-Y	SHEAR IN Y	BEND MOM-Z
1									
ELEMENT NO.	1	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-R0000+004
1	.200	.000	.000	.00000	.60000	.00000	.00000	.99999+003	-78000+004
1:1	.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-76000+004
1:2	.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-74000+004
1:3	.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-72000+004
1:4	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-70000+004
2L	1.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-70000+004
2R	1.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-68000+004
1:5	1.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-66000+004
1:6	1.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-64000+004
1:7	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-62000+004
1:8									-60000+004
2									
2									
ELEMENT NO.	2	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-60000+004
3	.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-58000+004
2:1	.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-56000+004
2:2	.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-54000+004
2:3	.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-52000+004
4L	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-50000+004
4R	1.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-50000+004
3:5	1.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-48000+004
3:6	1.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-46000+004
3:7	1.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-44000+004
3:8	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-42000+004
5									-40000+004
3									
ELEMENT NO.	3	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-40000+004
5	.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-38000+004
3:1	.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-36000+004
3:2	.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-34000+004
3:3	.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-32000+004
6L	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-30000+004
6R	1.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-30000+004
3:5	1.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-28000+004
3:6	1.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-26000+004
3:7	1.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-24000+004
3:8	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-22000+004
7									-20000+004
4									
ELEMENT NO.	4	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-20000+004
7	.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-18000+004
4:1	.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-16000+004
4:2	.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-14000+004
4:3	.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-12000+004
4:4	1.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-10000+004
8L	1.200	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-10000+004
8R	1.400	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-8000+003
4:5	1.600	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-6000+003
4:6	1.800	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-4000+003
4:7	2.000	.000	.000	.00000	.00000	.00000	.00000	.99999+003	-2000+003
4:8									-14648-002
4:9									

EMERGENCY ANALYSIS TO BE PERFORMED IMMEDIATELY!

RRRRRRR R RR
RR RR RR RR
RR RR RR RR
RR RR RR RR
RR RR RR RR

pppppppp p p p
pppppppp
pp p p p p
pp p p p p
pp p p p p
pp p p p p
pp p p p p

NO SUMMARY FOR FA. YET

GALLAGHER'S SQUARE SECTION TAPERING CANTILEVER - FOUR NONPRI

ROTATORY INERTIA EFFECTS TO BE INCLUDED IN ANALYSIS.

INERTIA ASSOCIATION

CODE:
0 INDICATES INERTIA NOT ASSOCIATED WITH GLOBAL D.O.F.
1 INDICATES INERTIA IS ASSOCIATED WITH GLOBAL D.O.F.
010001

MASS MATRIX TO BE EVALUATED BY SPECIAL LUMPING BASED ON CONSISTENT MATRIX.

LIMITS OF INTERVAL OF NATURAL FREQUENCIES TO BE CONSIDERED:

LOWER LIMIT=	1.0000	RAD./SEC.
UPPER LIMIT=	10000.	RAD./SEC.

ALL NATURAL FREQUENCIES LYING IN SPECIFIED INTERVAL ARE TO BE CALCULATED.

MODE SHAPES TO BE CALCULATED AND NORMALISED AT D.O.F. OF LARGEST MODULUS.

NO.OF FURTHER BISECTIONS TO BE PERFORMED ONCE A ROOT HAS BEEN ISOLATED= 6

RESULTS AT VARIOUS STAGES OF CONVERGENCE TO BE PRINTED OUT.

*** DYNAMIC CONDENSATION ROUTINE ENTERED!***

*** MATRIX COMPACTION ROUTINE ENTERED!***

*** SOLUTION ROUTINE ENTERED!***

ARRAYS ILU AND WLU

PROVIDES INFORMATION ON ISOLATED ROOTS

NO. OF ROOTS OF VALUE LESS THAN P (LOWER)	NO. OF ROOTS OF VALUE LESS THAN P (UPPER)	P (LOWER)	P (UPPER)
15	16	.30528-004	.61045-004
14	15	.38247-005	.76394-005
13	14	.96367-006	.19173-005
12	13	.48684-006	.96367-006
11	12	.24842-006	.48684-006
10	11	.69605-007	.12921-006
9	10	.39802-007	.69605-007
8	9	.10000-007	.39802-007

ARRAYS ILU AND WLU

PROVIDES INFORMATION ON ISOLAT D ROOTS AFTER NBIST TIMES FURTHER BISECTION

NO. OF ROOTS OF VALUE LESS THAN P (LOWER)	NO. OF ROOTS OF VALUE LESS THAN P (UPPER)	P (LOWER)	P (UPPER)
15	16	.54846-004	.55323-004
14	15	.67453-005	.68049-005
13	14	.15001-005	.15150-005
12	13	.60605-006	.61350-006
11	12	.25587-006	.25959-006
10	11	.11431-006	.11524-006
9	10	.54238-007	.54703-007
8	9	.29558-007	.30023-007

.54895098-004	.54895609-004	-.1065894940173+077	.7524138499683+075	.7524138499683+075
.67465818-005	.67465924-005	.2753488425705+062	-.1356825234369+061	-.1356825234369+061
.15027102-005	.15027170-005	-.1584613124631+053	.1541455118218+053	.1541455118218+053
.61254934-006	.61255183-006	.1451140064776+048	-.4531996569966+046	-.4531996569966+046
.25928875-006	.25928876-006	-.3149627632628+041	.8219759948776+041	.8219759948776+041
.11506336-006	.11506338-006	.2110275083001+037	-.1359957110941+038	-.1359957110941+038
.54638829-007	.54638831-007	-.7492141168100+033	.1266528749402+034	.1266528749402+034
.29904316-007	.29904334-007	.5154375997672+032	-.4479302121952+032	-.4479302121952+032

LIST OF EIGENVALUES

NUMBER	LAMBDA(P)	OMEGA (RAD./SEC)	OMEGA (CYCLES/SEC)
1	.54895609-004	.13496812+003	.21480861+002
2	.67465924-005	.38499737+003	.61274286+002
3	.15027170-005	.81575811+003	.12983204+003
4	.61255183-006	.12776991+004	.20335229+003
5	.25928876-006	.19638493+004	.31255658+003
6	.11506338-006	.29480268+004	.46919343+003
7	.54638831-007	.42780839+004	.68087878+003
8	.29904334-007	.57827302+004	.92035088+003

EIGENVECTOR FOR ROOT NO. = 1

EIGENVALUE = .54895609-004

1	10	7							
8	10	14	.0112	.0458	.0465	.1086	.0770	.2054	
15	10	16	.1137	.1551	.5213	.2012	.7466	.2418	
			1.0000						

EIGENVECTOR FOR ROOT NO. = 2

EIGENVALUE = .67465924-005

1	10	7							
8	10	14	-.0265	-.0900	-.0728	-.1751	-.0782	-.2401	
15	10	16	-.0364	.0760	-.0498	.2891	.3731	.5273	
			1.0000						

EIGENVECTOR FOR ROOT NO. = 3

EIGENVALUE = .15027170-005

1	10	7							
8	10	14	.0347	.0916	.0458	.1187	-.0091	.0564	
15	10	16	-.1131	-.1545	-.2089	-.0074	-.0351	.3796	
			.6535						

EIGENVECTOR FOR ROOT NO. = 4

EIGENVALUE = .61255183-006

1	10	7	-.0190	-.0177	-.0383	-.0057	-.0241	-.0303	.0270
8	10	14	.0559	.0611	-.0083	-.0175	-.1429	-.1319	.0578
15	10	16	.3551	1.0000					

EIGENVECTOR FOR ROOT NO. = 5

EIGENVALUE = .25928876-006

1	10	7	.0863	.0515	.1060	-.0634	-.0409	-.1332	-.1316
8	10	14	.0228	.0406	.1939	.1653	-.0522	-.1212	-.2193
15	10	16	.1708	1.0000					

EIGENVECTOR FOR ROOT NO. = 6

EIGENVALUE = .11506338-006

1	10	7	-.0527	-.0073	-.0079	.0868	.0596	-.0112	-.0642
8	10	14	-.1620	-.0604	.1453	.2197	.2337	-.0801	-.3808
15	10	16	.0744	1.0000					

EIGENVECTOR FOR ROOT NO. = 7

EIGENVALUE = .54638831-007

1	10	7	.0687	-.0341	-.0864	-.1352	-.0193	-.1411	.1586
8	10	14	-.0340	-.1031	-.1213	.1727	.5720	-.0455	-.4454
15	10	16	.0310	1.0000					

29904334-007

1	10	7	-	.0979	.1164	.2513	.0617	-.1385	-.1191	-.2416
2	10	14	.	.4532	-.0879	-.3399	.1087	.8128	-.0253	-.4643
15	10	16	.	.0139	1.0000					

ANALYSIS TERMINATED NORMALLY.

DPRT,S SADALE*NONPRI,SQCAN14ELNP
FURPUR 28R1 E35 S74T11 10/07/80 12:52:50

GALLAGHER'S SQUARE SECTION TAPERING CANTILEVER - FOUR NONPRI ELEMENTS

0 1 2 3 4 5
1 3 5 7 9 10
0 2 4 6 8 10
0 0 0 0 0 1
0 0 0 0 0 1
0 0 0 0 0 0

• 8333
• 6753
• 5333
• 4083
• 30833
• 13333
• 0753
• 03333

[illegible]

1	1	1	1	1	1	0	0.
1	1	1	1	1	1	3.	3E-3
.	.81	.	.64	.	.54	.	37SE-3
.	.64	.	.49	.	.40	.	13SE-3
.	.49	.	.36	.	.20	.	08SE-3
.	.36	.	.25	.	.10	.	AE-3
.	.25	.	.16	.	.05	.	20SE-3
.	.16	.	.09	.	.02	.	17SE-3
.	.09	.	.04	.	.	.	13EE
69.E9	26.E9	2.	7E3				

• • • • •
00000
x53210000
100009

POINT LOAD AT TIP(-1000)

2-1000.

END NO YES YES I O
END NO YES YES I O I

010001
21. 10000.

1. 1000.

0
1-5-1

QRRKPT PRINTS

Courses Completed in Partial Fulfillment
of the M.Sc.(Eng.) Degree at the University of Cape Town

<u>Course</u>	<u>Date Credited</u>	<u>Credit Value</u>
AM 318 NAC(a) Numerical Analysis and Computation	1979	5
AM 328 NAC(b) Numerical Analysis and Computation	1979	5
CE 515 Surface Structures	1979	5
CE 524 Structural Dynamics	1979	3
CE 535 Engineering Economy	1979	3
	Total	<u>21</u>

Course Credits : 21

Thesis Credits : 20

Total 41

Total credit requirements for the M.Sc.(Eng.) Degree : 40

UNIVERSITY OF CAPE TOWN
DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY EXAMINATION - DECEMBER 1979
CE 515 Surface Structures

Answer ALL questions

Time allowed: 3 hours

Notes and reference books are permitted

1.

Answer the following questions clearly and concisely.

- a. What are the advantages and disadvantages of the isoparametric formulation of element stiffness matrices, over the generalised coordinate method. (7)
- b. What are the characteristics of the shape functions for isoparametric elements. (7)
- c. If the total potential energy of a thin plate in bending is:

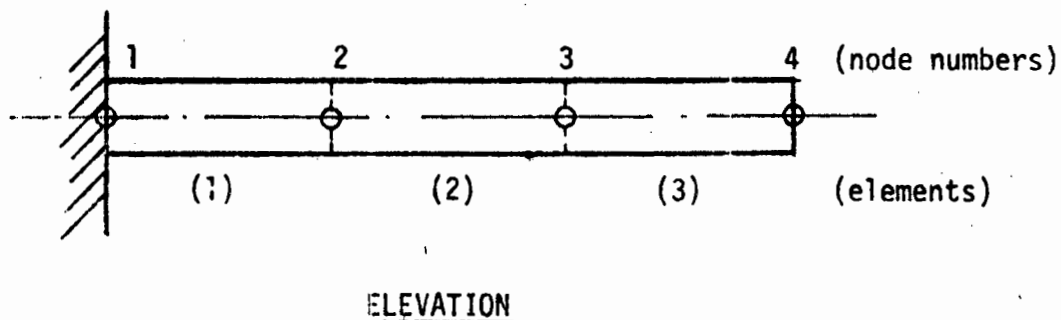
$$\Pi = \frac{1}{2} \int_{Vol} \{M\}^T \{X\} dV - \int_{Area} qwdA$$

where

$\{M\}$ is the vector of moments
 $\{X\}$ is the vector of bending deformation
 q is the lateral loading
 w is the lateral deflection

Show how an element stiffness matrix is formulated. (8)

- d. Evaluate the stress/strain relationship matrix for plane stress if Poissons ratio is 0,2 and isotropic materials are used. How will this matrix change if anisotropic materials are used. (7)
- e. What is the purpose of the Jacobian matrix used in the formulation of isoparametric element stiffness matrices. Use the Jacobian matrix for plate bending elements as an example in your explanation. (8)
- f. Use the following simple example to explain how element stiffness matrices can be used to calculate nodal displacements with respect to a set of global axes.

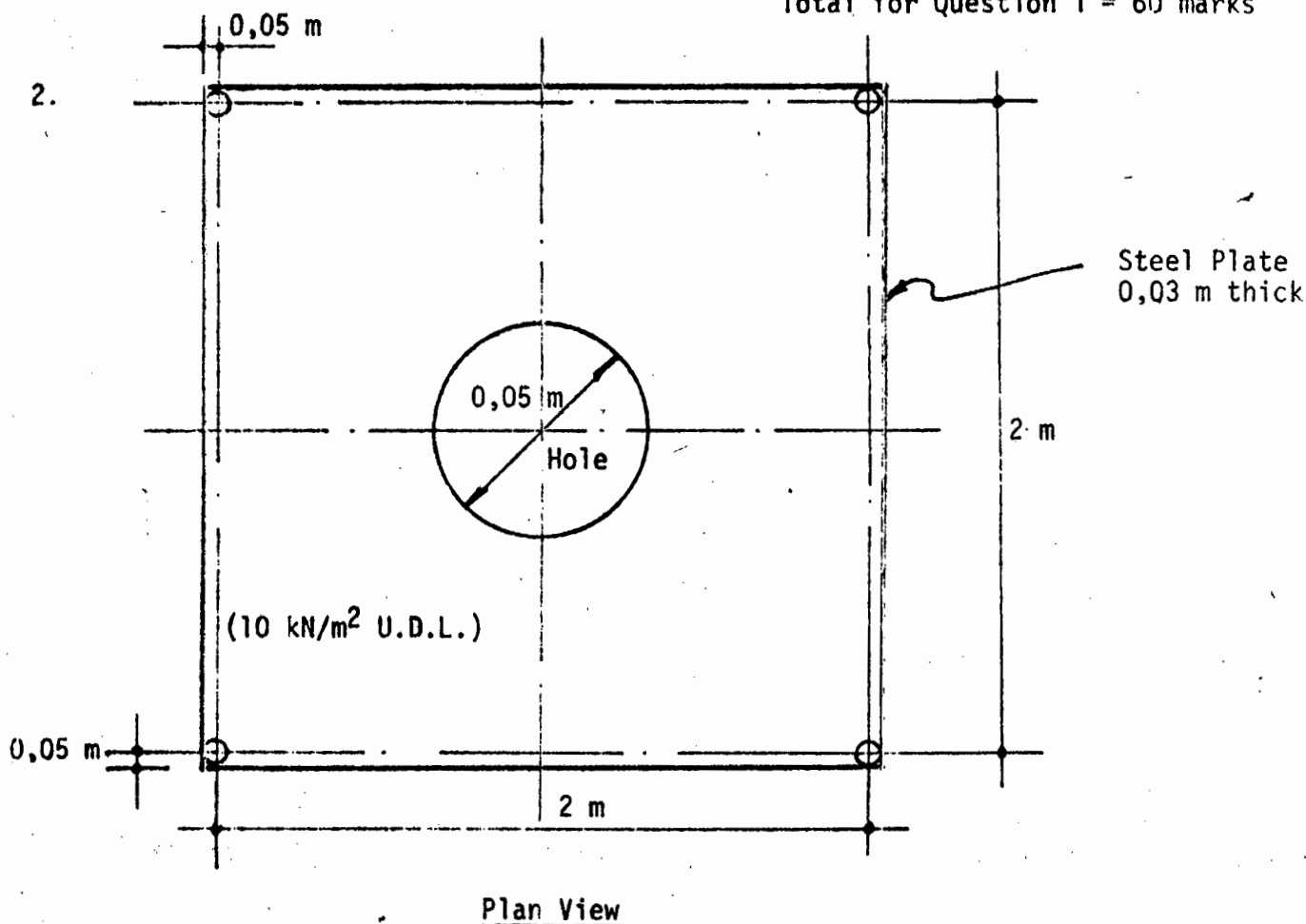


Cantilever beam assumed to consist of 3 discrete two noded elements (8)

- g. Is it true to say that a banded solution technique requires less immediate access storage in a computer than a frontal solution technique. Explain the differences between the two techniques. (8)
- h. Use Gauss quadrature to integrate the following function (use a 2 point rule)

$$\int_{-1}^1 S\left(\frac{2}{3}\xi - \frac{4}{3}\xi^2\right) d\xi \quad (7)$$

Total for Question 1 = 60 marks



The steel plate with a central hole, shown overleaf, is supported on four corner columns and subjected to a uniformly distributed vertical load of 10kN/m^2 .

- a. By using the data preparation instructions attached, write out the data required for the finite element solution of the plate.
(Use a sufficient number of elements to ensure that the solution will be reasonably accurate).

(35)

- b. What detailed information can be expected from the finite element analysis of this type of problem.

(5)

Total for Question 2 = 40 marks

UNIVERSITY OF CAPE TOWN
DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY EXAMINATION - NOVEMBER 1979
CE 524 STRUCTURAL DYNAMICS

INSTRUCTIONS

This examination is a 'take home' exam. The examination paper may be picked up from noon 23 November 1979 and the script containing your answer must be returned before noon on Monday, 3 December 1979.

You may consult text books, papers and notes, but the question must not be discussed with any other person. A declaration to this effect must accompany the script on submission.

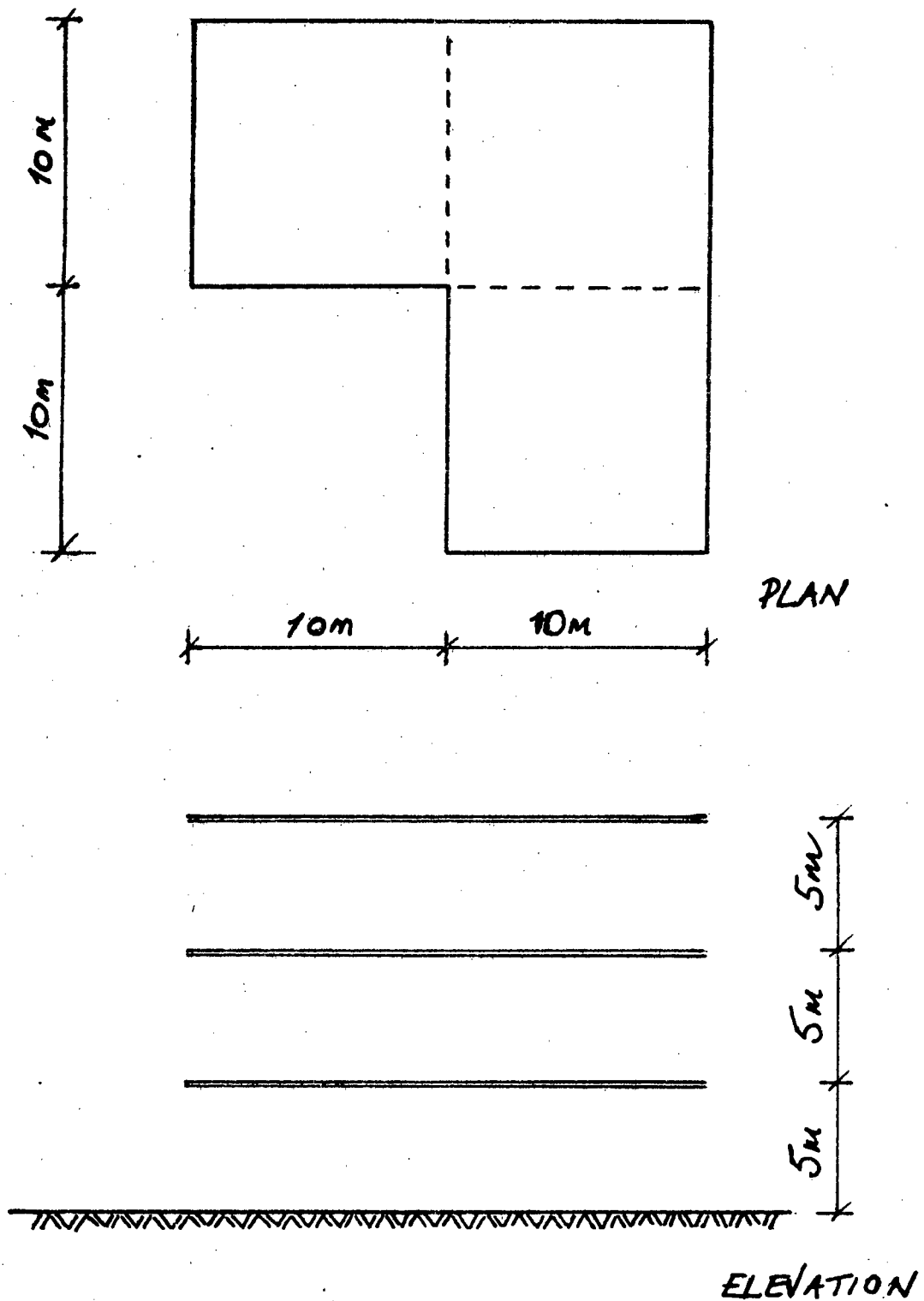
Mr P.B. Griffin will be available from 9.00 am to 12 noon on Saturday, 24 November and on Saturday, 1 December, and during the week of 26 November by appointment, to run programs discussed during the course with data provided by the candidates.

QUESTION

A three storey disaster control centre is to be constructed in Cape Town. The building, with overall dimensions, is shown in the attached diagram. Each floor can be considered to be rigid and to have a mass (including fixed equipment) of 150 000 kg. Steel columns are to be used, but the spacing, layout and orientation of the I section columns has not yet been decided. Columns may be located, however, only along the outer perimeter of the floor slab and along the dotted lines shown on the plan.

Because of the nature of its use, the building must remain operational under earthquake loading. Consequently the columns are to be designed to behave elastically (with a maximum stress not exceeding 240 MPa) under dead load plus an earthquake whose design spectrum is given in Example 10.3 in the course notes. 5% damping may be assumed.

Give a suitable layout and sizing of columns to meet this design condition. Assume that the mass of the slabs is uniformly distributed, and allocate the weight of the slabs to the various columns by any reasonable assumption. Give reasons why you have chosen the layout you have used, and if you believe that improvement on your design is possible describe how this might be accomplished.



PROJECTED DISASTER CONTROL CENTRE FOR
CITY OF CAPE TOWN

DEPARTMENT OF CIVIL ENGINEERING

UNIVERSITY OF CAPE TOWN

CE 535

ENGINEERING ECONOMY

FINAL EXAMINATION

1979

3 HOURS

OPEN BOOK

ANSWER ALL QUESTIONS

CALCULATORS MAY BE USED

- (1) You have two mutually exclusive investment opportunities, both of which have a life of 3 years with no salvage value. Both cost R120,000 and produce revenues as shown in Table 1. Bob, one of your analysts, has calculated the NPV at your cost of capital (10%) and selects B on the grounds of a higher NPV. Andy, another analyst, has selected A on the grounds of a higher IRR.

- (a) Are they correct?
- (b) If so, why do they differ?
- (c) What is your recommendation.

TABLE 1.1

	<u>INVESTMENT</u>	<u>INVESTMENT</u>
	<u>A</u>	<u>B</u>
Cost	120,000	120,000
Revenue Year 1	100,000	10,000
Year 2	50,000	60,000
Year 3	10,000	110,000

(12 MARKS)

- (2) Your Swazi client has negotiated a E6,000,000 interest free loan to construct a teaching complex and has appointed you to head the project management team. An architect has been appointed to design the facility and you are to provide him with a budget figure within which to work.

You expect the design phase to take 9 months, documentation and tendering 3 months and construction 12 months. The lending authority will make payment against your interim valuations as the work progresses, but will under no circumstances disburse more than the value of the loan.

- (a) Make and motivate all necessary assumptions.
 - (b) Calculate the design budget figure on the basis that the architect's knowledge of costs is probably 3 months old.
- (3) A major employer is proposing to introduce the escalation formula described in Appendix 1.

Critically examine:

- (a) The differences between the proposed formula and other existing formulae.
- (b) The proposed indices, their applicability and desirability.
- (c) The proposed coefficients.
- (d) The proposals potential for accuracy relative to other formulae.
- (e) The need and desirability for a major employer to develop his own formula.

(20 MARKS)

- (4) The life of an item of equipment may be described in any of the following ways:

- Ownership life.
- Physical life.
- Primary service life.
- Accounting life.
- Tax life.
- Economic life.
- Useful life.

Define each period with particular regard to the manner in which the period is assessed and/or the factors which serve to terminate the period.

(12 MARKS)

- (5) A proposal has been made that a new piece of equipment be purchased this year. Characteristics of the purchase plan are given in Table 5.1. Calculate the before tax IRR. (use $i = 12$ and 15). If the effective tax rate is 40% and if wear and tear allowances are granted on a straight line basis over 5 years, calculate the after tax IRR. (use $i = 7$ and 10).

TABLE 5.1

P	=	R50,000	
SV	=	0	
n	=	5 years	
Expected income	=	28,000 - 1,000k	(k = 1, 2, 3, 4, 5)
Expected disbursements	=	9,500 + 500k	

(12 MARKS)

- (6) One phase of a meat-packing operation requires the use of separate machines for the following functions: pressing, slicing, weighing, and wrapping. All machines under consideration are expected to have a life of six years with no salvage value. Their first cost and annual costs are given in Table 6.1.
- (a) If different machines can be selected from different manufacturers and if the company's Minimum Attractive Rate of Return is 20% , determine which separate machine should be selected for each function (identify them as Pressing 1, Pressing 2, Slicing 1, etc.)
- (b) For the machines selected in part (a), determine the total investment and operating cost for the entire operation.

Another alternative is one large machine to do the pressing and slicing and another large machine to do the weighing and wrapping. The machine that will do the pressing and slicing (identified as Pressing-Slicing 3) will cost R29,000 and will have an annual operating cost of R9,000. The machine that will do the weighing and wrapping (identified as Weighing-Wrapping 3) will cost R26,000 and will have an annual operating cost of R18,000.

- (c) Which machines should be selected for the entire operation?
- (d) Determine the total investment and operating cost for the entire operation.
- (e) If a single machine is to perform all four functions (identified as Machine 4) having an initial cost of R45,000 and an annual operation cost of R32,000 is available - which machine(s) should be selected?

TABLE 6.1

	<u>SUPPLIER 1</u>		<u>SUPPLIER 2</u>	
	<u>First cost</u>	<u>Annual cost</u>	<u>First cost</u>	<u>Annual cost</u>
Pressing	R5,000	R13,000	R10,000	R11,000
Slicing	4,000	10,000	17,000	4,000
Weighing	12,000	15,000	15,000	13,000
Wrapping	3,000	9,000	11,000	7,000

(20 MARKS)

- (7) The Cash Flow approach to capital consumption cost recognises an outflow (C) at the end of year 0 and an inflow (C_A) at the end of the assets life (A yrs). The annual capital consumption cost thus becomes $C \cdot (A/P i_A) - C_A \cdot (A/F i_A)$

Establish the relationship between this expression and expressions which provide for a sinking fund plus profit on initial capital or which provide for capital recovery plus interest on residual value.

(12 MARKS)

UNIVERSITY OF CAPE TOWN

UNIVERSITY EXAMINATION : NOVEMBER 1979

Numerical Analysis and Computation (a)

AM318 and AM346

Time : 3 hours

Not more than FIVE questions to be answered :

1.(a) The iterative procedure

$$x_{r+1} = F(x_r)$$

to find an approximate solution of the equation

$$x = F(x)$$

is known as the 'method of successive substitutions'. Discuss the convergence of this procedure, and illustrate by finding a convergent procedure to solve the equation

$$x + \log_e x = 0 .$$

Taking $x_0 = 0,5$, find a solution to three decimal places, and check your solution by applying the Newton-Raphson iterative method.

(b) When $f(x)$ has two zeros α_1 and α_2 which are nearly coincident, so that $f'(x)$ is zero at a point β between α_1 and α_2 , show that, if β is found, approximations to α_1 and α_2 are given by

$$\alpha = \beta \pm \left\{ - \frac{2f(\beta)}{f''(\beta)} \right\}^{\frac{1}{2}} .$$

Improved values may then be obtained by the usual methods.

Hence determine approximately the nearly coincident real roots of the equation

$$3x^4 + 8x^3 - 6x^2 - 24x + 18 = 0 .$$

University of Cape Town, University Examination, November 1979, Numerical Analysis and Computation (a) continued:

2. Given the tabulated function

x	:	1	2	3	4	5	6
f(x)	:	2439	2174	1961	1786	1639	1515

(a) Draw up a difference table

(b) Find $f(4,2)$ using Stirling's interpolation formula

$$f(x) = f_0 + \theta \mu \delta f_0 + \frac{1}{2!} \theta^2 \delta^2 f_0 + \frac{1}{3!} \theta (\theta^2 - 1) \mu \delta^3 f_0 + \frac{1}{4!} \theta^2 (\theta^2 - 1) \delta^4 f_0 + \dots$$

(c) Find $f'(3)$ by differentiating Stirling's formula

(d) Find $\int_1^5 f(x) dx$ by integrating Stirling's formula.

Show the relation of the quadrature formula obtained to Simpson's rule.

(20)

3. A two-point Gaussian Quadrature formula has form:

$$\int_{-h}^h f(x) dx = h \{ a f(\alpha h) + b f(\beta h) \} + E$$

Evaluate the weighting constants a and b and the position parameters α and β so that the formula is exact ($E = 0$) for $f(x)$ any polynomial of degree ≤ 3 .

Find E when $f(x) = x^4$

Calculate $\int_{-0,5}^{0,5} x^4 dx$

- (i) using the Gaussian Quadrature formula
- (ii) analytically
- (iii) using Simpson's rule (see page 3)

continued on Page 3.

University of Cape Town, University Examination, November 1979,
Numerical Analysis and Computation (a) continued:

3. (continued):

$$\int_{x_0}^{x_0+2h} f(x) dx \approx \frac{h}{3} \{ f(x_0) + 4f(x_0+h) + f(x_0+2h) \}$$

Show that if $f(x)$ has a Taylor series expansion about $x = 0$ then

$$E \approx \frac{h^5}{135} f^{(5)}(0) . \quad (20)$$

4. Describe the Taylor series method for the approximate solution of the differential equation

$$\frac{dy}{dx} = f(x, y) \quad ; \quad y(0) = y_0$$

Show that the Simple Runge-Kutta formula

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{2}(k_1 + k_2) \\ k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + h, y_n + k_1) \end{aligned}$$

provides an approximation to the Taylor series method and find the order of the term in h to which there is agreement.

Compare the approximate solutions obtained by the Taylor series, Simple Runge-Kutta and Euler methods for two steps ($h = 0,1$) in the problem:

$$\frac{dy}{dx} = x^2 + y^2 \quad y(0) = 1 \quad (20)$$

5. (a) For the set of equations

$$\begin{aligned} 2x_1 + 5x_2 - x_3 &= 7 \\ 4x_1 + x_2 + x_3 &= -1 \\ -x_1 - x_2 + 3x_3 &= 2 \end{aligned}$$

(i) Describe briefly three variations of the Gaussian elimination method, and solve by the complete pivoting method.

University of Cape Town, University Examination, November 1979
Numerical Analysis & Computation (a) continued:

- 5.(ii) Show that, for one re-arrangement of these equations, the Gauss-Seidel iterative method will work, but for the equations as they stand it will not. Perform two iterations in each of these cases, and compare the results with the exact solution $(-1; 2; 1)$.

(b) For the set of equations

$$1,7 x_1 + 2,3 x_2 - 1,5 x_3 = 2,35$$

$$1,1 x_1 + 1,6 x_2 - 1,9 x_3 = -0,94$$

$$2,7 x_1 - 2,2 x_2 + 1,5 x_3 = 2,70$$

it has been found that there is a solution near $(1; 2; 3)$.

Find an improved solution.

(23)

6. A firm manufactures three types of electrical device: A, B and C on which it makes a gross profit of 40c, 30c and 45c per device, respectively. Each type A device requires twice as much machine time as a type B device and 80 per cent as much time as a type C device. If only type A devices were manufactured, there would be enough machine time per day to make exactly 1 200 of them. The supply of copper wire is sufficient for only 1 000 devices to be made each day (all types combined). The type B device requires a ceramic insulator of which only 700 are available per day, and the highly-skilled labour needed to produce type C devices permits no more than 400 of these devices to be produced each day.

Find the production plan which maximises the gross profit subject to these limitations.

(20)

- 7.(a) For a tabulated function $f(x)$ it is desired to approximate to $f(x)$ in the least-square sense by a polynomial of degree m . Show that this approximation leads to a system of linear equations.

Find the least-square parabola for the function

x	0	0,1	0,2	0,3	0,4
$f(x)$	0,0	0,0998	0,1987	0,2955	0,3894

Describe briefly how a value obtained from this parabola would differ from a value obtained by finite difference interpolation.

University of Cape Town, University Examination, November 1979,
Numerical Analysis and Computation (a) continued:

- 7.(b) Discuss the approximation to $y(x)$ by orthonormal functions $\phi_r(x)$

$$y = \sum_r a_r \phi_r(x)$$

and give an expression for the a_r .

Consider the particular case of Fourier series, and find a Fourier expansion for

$$g(x) = x \quad -\pi < x < \pi$$

(24)

- 8.(a) Explain how you would find the eigenvalue of maximum modulus of a real, symmetric matrix, and why the process converges.

- (b) For the eigenvalue problem $(\underline{A} - \lambda \underline{I})\bar{x} = 0$ given that

the matrix $\underline{A} = \begin{pmatrix} 7 & 3 & 3 & 0 \\ -3 & -2 & 2 & -3 \\ 5 & 3 & 1 & -2 \\ 5 & 3 & -3 & 2 \end{pmatrix}$

has eigenvalue $\lambda = 4$ with associated eigenvector

$(1 ; -1 ; 0 ; 1)$ find using the orthogonality property

$\bar{x}_i^T \bar{x}_j = 0$ a 3×3 matrix \underline{B} from which the other eigenvalues of \underline{A} could be determined.

What would be an alternative method of finding another eigenvalue?

(22)